

# BOOK of GAMES

COMPUTER  
& VIDEO  
GAMES

FLESHscan

TERMINATE+++++

ANDROID  
CONFIRMED  
TARGET  
LOCKED+++

# CONTENTS

Cover by  
Jeff Riddle

## A WORD FROM



## THE EDITOR

Hi there, gamers, it's time for another bumper Book of Games just in time to stop you getting bored during the summer holidays.

We think we've come up with another interesting and entertaining selection of programs sent in by our readers.

Besides being fun to play, they'll hopefully help you improve your programming skills.

All the games have been tried and tested by our team of reviewers. Bugs should be few and

far between — but if you have any problems, don't hesitate to call C&VG's Bug Hunter who will try and solve your problems.

And if these listings have inspired you to try your hand at coming up with a game of your own, we may be able to publish it.

### METEOR CBM 64

The star fleet has been destroyed and your ship is the only survivor. Now a meteor storm strikes. Can you make it back to the Mother Ship?

4

### KEYCHASE BBC

Find the scattered keys to unlock the door to success. But beware the Pacmen who are bent on your destruction.

8

### BREAKOUT UNEXPANDED VIC 20

Another brick in the wall... and a whole lot of points to be won by breaking out.

10

### SPECTRANAUT AMSTRAD

Your space shuttle runs out of fuel and is forced to crashland on a hostile planet. But to collect more fuel you must brave the planet's perils.

12

### RED FRED SPECTRUM

Snakes Alive! Red Fred's in peril as he jumps from block to block painting his pyramid. Will the snake and the bouncing ball beat him?

15

### POWERBOAT ATARI

Oh Buoy! High speed action as you collect the yellow buoys but steer clear of the red ones.

18

### TIMEBOMB UNEXPANDED VIC 20

Time ticks on. It could be seconds before it explodes. But having defused one bomb, another appears. Can you keep your nerve?

20

### SKYWALKER BBC

Test your skills with a Light Sabre against deadly bolts of energy. Quick wits are needed in this skirmish.

22

### CATERPILLAR AMSTRAD

Creepy-crawly capers among the mushrooms. The more you eat, the more you score.

26

### SHOWJUMPING ATARI

Test your horse sense. Have you the skill to guide your trusty mount over the jumps?

28

### SEWER RATS CBM 64

Venture down into the rat-infested sewers. But watch out that they do not make a meal of you!

30

### ROCK FALL SPECTRUM

Diamonds may be forever but you won't be as you hunt these precious stones in a rock fall.

32

# SEND US YOUR GAMES!



Have you written a games program that you'd like to see printed within C&VG? You have? Then fill in this form and rush it to us at Computer & Video Games, Program Listings, Priory Court, 30-32 Farringdon Lane, London EC1R 3AU. We pay £25 for each listing we print — so get your game to us today!

We'd also like you to send us a photograph of yourself — it doesn't matter how silly you look — so we can feature your smiling face alongside your listing and make you REALLY famous. Money and fame — what more could you ask for?

Please fill in the form below carefully so we know exactly what's going on in your game. And please include full instructions and program details along with your tape. Don't forget to keep a copy of your game for yourself. We're afraid that we can't return your tapes to you.

Talking of tapes — that's how we'd like you to send your games in to us. On tape or disc — with a listing if you have a printer.

Lots of our readers have been asking just what happens to their games when they arrive at C&VG. Well, we send them out to our team of hard working reviewers who let us know just what they think of your game. If they like it, your tape will go into our

"to be published" files. And that's where the waiting starts. We receive several hundred tapes each month — and loads of them are printable. So it can take quite a while to see your game in print. But we like to think that it's worth the wait!

And starting here and now you could become C&VG's Programmer of the Year — which we'll judge at Christmas time. So you could get an unexpected Christmas bonus. And don't worry those of you who already have had programs printed — you'll be eligible too!

Program name: .....

Machine make: ..... Model: .....

Other models it should run on: ..... Number of K needed to run it: .....

Other equipment (joysticks, Extended Basic, add ons, etc.) needed to run it: .....

Author's Christian name: ..... Surname: .....

Address: .....

..... Tel: ..... Date: .....

Type of game: (If original please say so) .....

Loading instructions: .....

Game instructions: (If not included in the listing) .....

## Office use only

Date received: .....

Acknowledgement sent: ☐ Evaluator's comments

Name of evaluator: ..... Good enough to publish ☐

Date sent out: ..... Needs some tidying up ☐

Date due back: ..... Not worth publishing ☐

Needs to be returned to author for alterations: ☐ Date sent: ..... Same game already published on this micro ☐

Due to be published in ..... Wouldn't load ☐  
issue of magazine.

# METEOR BY ALLAN SHORTT

All weapons are out, the rest of the fleet is destroyed, and you're stuck in the middle of a meteor storm, desperately trying to make it to the Mother Ship. I know what you're thinking, "Why does it always happen to me?!"

The idea of the game is to dodge the approaching chunks of rock to dock with the Mother Ship at the top of the screen. The keys are:

Cursor Right = RIGHT  
Cursor Down = LEFT  
Space Bar = UP

When typing in the listing, you'll often find words within square brackets, indicating the keys you should press. This method eliminates all those nasty-looking graphics characters. So:

[HOME] — Press the HOME key.  
[RED] — Colour Red.  
[sX] — Press key X whilst pressing SHIFT.  
[cX] — Press key X whilst pressing C=.

```
41 HIS=0
42 R1=10:R2=13:R3=9:R4=16:R5=14:R6=15:R7
=10
50 SOUND=10
60 ENX=9:SC=0:TR=0:U=55296:U1=1024
70 REM ** SET SOUND VARIABLES **
80 A1=54277:A2=54284:S1=54278:S2=54285:W
1=54276:W2=54283
90 H1=54273:H2=54280:L1=54272:L2=54279
110 POKE 53280,0:POKE 53281,0
120 PRINT "[CLEAR][WHITE][DOWN][DOWN][RI
GHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][
RIGHT][RIGHT][RIGHT]METEOR DODGER II"
130 PRINT "[WHITE][DOWN][DOWN][DOWN][DO
WN][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RI
GHT]WRITTEN BY ALLAN SHORTT"
135 FOR Q=1 TO 500:NEXT Q
140 PRINT "[DOWN][DOWN][DOWN][DOWN][DOWN
][DOWN][DOWN][DOWN][DOWN][DOWN][DOWN][
DOWN][DOWN][RVSON][RIGHT][RIGHT][RIGHT][RI
GHT][RIGHT][RIGHT]PRESS SPACE BAR TO STA
RT"
150 FOR A=55385 TO 55406
160 POKE A,INT(12*RND(1)+1)
170 GET A$:IF A$="" THEN NEXT A
180 IF A$="" THEN 210
190 FOR Z=1 TO 20:NEXT
200 GOTO 150
210 REM ** SET UP VARIABLES FOR ROCKS **
*
230 GOSUB 990
240 GOSUB 1060
250 GOTO 1630
260 REM ** PRINT UP SCORE AND ENERGY **
270 POKE 53280,0:POKE 53281,0
280 PRINT"[CLEAR]";
290 PRINT "[HOME][RED][RVSON]
";
300 PRINTTAB(30):"[RVSOFF][WHITE]ENERGY=
":ENX="";
310 PRINT "[c C][c Y][c U][c *][RVSON]
HOME [RVSOFF][sEP][c Y][c T][
c Y]"
320 PRINT " [c T][c T][c T][c Y][
c Y][c U][c U][c *][c U][c U][c U][
sEP][c U][c U][c U][c Y][c Y][c T][c T][
c T]";
330 PRINT TAB(30):"[WHITE][DOWN][DOWN][D
OWN]SCORE":SC="";
331 PRINT TAB(30):"[DOWN]TRIP":TR
335 PRINT TAB(30):"[DOWN][DOWN][DOWN][DO
WN][DOWN][DOWN]HI ":"HIS
340 REM ** POSITION SPRITES ON SCREEN **
350 X1=100:X2=200:X3=230:X4=150:X5=30
```

```
355 R1=RND(100):R2=RND(100):R3=RND(100):
R4=RND(100):R5=RND(100):R6=RND(100):
356 R7=RND(100)
360 X6=110:R6=30:X7=220:R7=30
370 X=127:Y=220
380 REM ** COLOUR FOR ALL SPRITES ***
390 POKE V+29,130
400 POKE V+23,130
410 POKE V+39,1
420 POKE V+40,13
430 POKE V+41,8
440 POKE V+42,3
450 POKE V+44,8
460 POKE V+45,13
470 POKE V+46,7
480 REM ** CHECK FOR KEYPRESSED **
490 GET A$:IF A$="" THEN 580
500 IF A$="[DOWN]" THEN X=X-10
510 IF A$="[RIGHT]" THEN X=X+10
520 IF A$="" THEN Y=Y-3
530 IF X<30 THEN X=30
540 IF X>250 THEN X=250
550 IF Y<65 AND X>100 AND X<155 THEN G
OSUB 1830
560 IF Y<65 THEN Y=65
570 REM ** SPEED OF EACH SPRITE ROCK **
580 R1=R1+9:G:R2=R2+12:G:R3=R3+15:R4=R4+
10:R5=R5+9:R6=R6+11:G:R7=R7+12
```





# C8764

```

581 POKE U1,194:POKEU1-1,160
582 U1=U1+1
584 IF U1>=1053 THEN U1=1024
590 REM ** SEE IF SPRITE HITS BOTTEM **
600 IF R1>=255 THEN R1=30:SC=SC+10:GOSUB
1540:GOSUB1190:POKEV+2,X1:POKEV+3,R1
610 IF R2>=255 THEN R2=30:GOSUB 1540:GOS
UB 1200:POKEV+4,X2:POKEV+5,R2
620 IF R3>=255 THEN R3=30:GOSUB 1540:GOS
UB 1210:POKEV+6,X3:POKEV+7,R3
630 IF R4>=255 THEN R4=30:GOSUB 1540:GOS
UB 1220:POKEV+8,X4:POKEV+9,R4
640 IF R5>=255 THEN R5=30:GOSUB 1540:GOS
UB 1230:POKEV+10,X5:POKEV+11,R5
650 IF R6>=255 THEN R6=30:GOSUB 1540:GOS
UB 1240:POKEV+12,X6:POKEV+13,R6
660 IF R7>=255 THEN R7=30:GOSUB 1540:GOS
UB 1250:POKEV+14,X7:POKEV+15,R7

```

```

670 REM ** CHECK FOR SPRITES COLLISION *
680 IF (PEEK(V+30)AND1)=1 THEN GOSUB 127
0:GOSUB 1320
690 IF (PEEK(V+30)AND1)=3 THEN GOSUB 127
0:GOSUB 1330
700 IF (PEEK(V+30)AND1)=5 THEN GOSUB 127
0:GOSUB 1340
710 IF (PEEK(V+30)AND1)=9 THEN GOSUB 127
0:GOSUB 1350
720 IF (PEEK(V+30)AND1)=17 THEN GOSUB 127
0:GOSUB 1360
730 IF (PEEK(V+30)AND1)=33 THEN GOSUB 127
0:GOSUB 1370
740 IF (PEEK(V+30)AND1)=65 THEN GOSUB 127
0:GOSUB 1380
750 IF (PEEK(V+30)AND1)=129 THEN GOSUB 127
0:GOSUB 1390
760 REM ** POSITION SPRITES ON SCREEN *
770 POKE V+21,255
780 POKE V+40,255
790 POKE V+0,X
800 POKE V+1,Y
810 POKE V+2,X1
820 POKE V+3,R1
830 POKE V+4,X2
840 POKE V+5,R2
850 POKE V+6,X3
860 POKE V+7,R3
870 POKE V+8,X4
880 POKE V+9,R4
890 POKE V+10,X5
900 POKE V+11,R5
910 POKE V+12,X6
920 POKE V+13,R6
930 POKE V+14,X7
940 POKE V+15,R7
950 POKE 54296,SOUND:POKE A1,128:POKE S1
128:POKE W1,129:POKE H1,5:POKE L1,R1

```

# METEOR BY ALLAN SHORTT

```

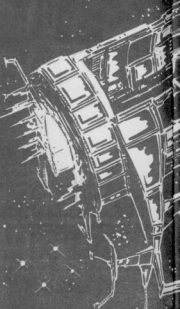
960 IF ENX<1 THEN 1110
980 GOTO 490
990 PRINT"[CLEAR]":
1000 REM ** MAKE UP EACH SPRITE ***
1010 V=53248
1020 POKE 2040,13
1030 FOR S=0 TO 62:READ D:POKE 832+S,D
1040 NEXT S
1050 RETURN
1060 POKE 2041,14:POKE 2042,14:POKE 2043
,14:POKE 2044,14:POKE 2045,14
1070 POKE 2046,14:POKE 2047,14
1080 FOR S=0 TO 62:READ D:POKE 896+S,D:N
EXT
1090 RETURN
1100 DATA 64,0,2,64,0,2,224,0,7,224,0,7,
160,0,5,160,0,5,160,24,5,160,24,5
1110 DATA 240,24,15,176,60,13,184,126,29
,184,91,29,190,91,125,254,126,127
1120 DATA 255,255,255,191,255,255,191,25
,9,253,223,255,253,231,143
1130 DATA 103,231,206,63,195,252
1140 DATA 3,224,0,3,254,0,7,255,0,15,255
,224,33,255,240,63,255,224
1150 DATA 63,255,224,63,255,252,127,255,
254,127,255,254,127,255,254
1160 DATA 127,255,252,127,255,252,63,255
,252,63,255,248,63,255,224
1170 DATA 30,255,192,6,127,224,0,63,128,
0,6,0,0,0,0
1180 REM ** POSITION SPRITE AT RANDOM **
1190 X1=INT(250*RND(1)+1):RETURN
1200 X2=INT(250*RND(1)+1):RETURN
1210 X3=INT(250*RND(1)+1):RETURN
1220 X4=INT(250*RND(1)+1):RETURN
1230 X5=INT(250*RND(1)+1):RETURN
1240 X6=INT(250*RND(1)+1):RETURN
1250 X7=INT(250*RND(1)+1):RETURN
1260 GOTO 490
1270 ENX=ENX-1
1280 IF ENX<1 THEN 1320
1290 PRINT "[HOME]":TAB(37):ENX
1300 RETURN
1310 REM ** MOVE SPRITE TO TOP OF SCREEN
1320 POKE V+0,X:POKE V+1,Y:RETURN
1330 POKE V+2,30:POKE V+3,20:RETURN
1340 POKE V+4,60:POKE V+5,20:RETURN
1350 POKE V+6,90:POKE V+7,20:RETURN
1360 POKE V+8,100:POKE V+9,20:RETURN
1370 POKE V+10,110:POKE V+11,20:RETURN
1380 POKE V+12,140:POKE V+13,20:RETURN
1390 POKE V+14,180:POKE V+15,20:RETURN
1400 IF ENX>=1 THEN RETURN
1410 REM ** GAME OVER ROUTINE **
1420 POKE 54296,0
1430 ZZ=1
1440 GOSUB 1840
1450 ZZ=0
1460 POKE 53280,0:POKE 53281,0
1470 PRINT "[CLEAR] GAME OVER "
1480 PRINT "[HOME]":
1490 PRINT "[WHITE][DOWN][DOWN][DOWN][RI
GHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][
RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT]
HT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][R
IGHT]YOU SCORED :":SC
1495 IF SC>HIS THEN HIS=SC
1500 PRINT "[DOWN][DOWN][DOWN][DOWN][DOWN]

```

```

N][DOWN][DOWN][DOWN][DOWN][DOWN][DOWN]
OWN][DOWN][DOWN][DOWN][DOWN][DOWN][DOWN]
LAY AGAIN "
1510 FOR A=1 TO 2000:NEXT
1520 GET A$:IF A$="" THEN 1520
1530 RESTORE:GOTO 42
1540 SC=SC+1
1550 PRINT "[HOME]":TAB(35):"[WHITE][DOWN]
1610 REM ** INSTRUCTIONS ***
1620 GOTO 260
1630 PRINT "[CLEAR][WHITE]":
1640 PRINT "[DOWN][DOWN]YOU ARE ABOUT TO
ENTER A ASTROID BELT"
1650 PRINT"WHERE YOU HAVE TO DODGE THE A
STROIDS IN"
1660 PRINT"ORDER TO SURVIVE. YOUR LAZE
RS HAVE BEEN":
1670 PRINT"SMASHED TO PIECES. SO HAS MO
ST OF YOUR "
1680 PRINT"SPACE SHIP. IF YOU REACH YO
UR PLANET"
1690 PRINT"YOU WILL SURVIVE BUT BEWARE
YOU SHALL"
1700 PRINT"SOON HAVE TO CONTENT AGAIN."
1710 PRINT "[DOWN][DOWN][DOWN]DO YOU WAN
T SOUND (Y/N)"
1720 GET A$:IF A$=""THEN 1720
1730 IF A$="Y" THEN GOTO 1750
1740 IF A$="N" THEN SOUND=0:GOTO 1750
1750 GOTO 1720
1750 PRINT "[DOWN][DOWN][DOWN][DOWN][DOW
N]PRESS ANY KEY TO SEE INSTRUCTIONS."
1760 GET A$:IF A$="" THEN 1760

```



# C64

```
1770 PRINT "[CLEAR][WHITE]";
1780 PRINT "[DOWN][DOWN][DOWN][RIGHT][RI
GHT][RIGHT][RIGHT][RIGHT][ CURSOR RIGHT
] = LEFT
1790 PRINT "[DOWN][RIGHT][RIGHT][RIGHT][R
IGHT][RIGHT][ CURSOR DOWN ] = RIGHT
1800 PRINT "[DOWN][RIGHT][RIGHT][RIGHT][
RIGHT][RIGHT][ SPACE BAR ] = UP"
1810 FOR A=1 TO 2000:NEXT
1820 GOTO 260
1830 REM MOVE SPRITES TO TOP OF SCREEN
1831 POKE 54296,15
1832 POKE A1,128:POKE S1,128:POKE W1,33
1833 FOR Z=1 TO 125
1834 POKE H1,Z:POKE L1,Z+50
1835 NEXT Z
1840 POKE 54296,SOUND
1850 POKE A2,128:POKE S2,128:POKE W2,129
```

```
1860 FOR T=1 TO 50
1870 C=INT(12*RND(1)+2)
1880 POKE H2,INT(6*RND(1)+1):POKE L2,INT
(200*RND(20)+10)
1890 POKE 53280,C:POKE 53281,C/2
1900 NEXT T
1910 POKE 54296,0:POKE A2,0:POKE S2,0:PO
KE W2,0
1915 SC=SC+(ENX*10)
1920 POKE V+0,0:POKE V+1,0
1930 POKE V+2,0:POKE V+3,0
1940 POKE V+4,0:POKE V+5,0
1950 POKE V+6,0:POKE V+7,0
1960 POKE V+8,0:POKE V+9,0
1970 POKE V+10,0:POKE V+11,0
1980 POKE V+12,0:POKE V+13,0
1990 POKE V+14,0:POKE V+15,0
2000 IF ZZ=1 THEN RETURN
2010 PRINT "[CLEAR]";
2015 TR=TR+1
2020 POKE 53280,6:POKE 53281,6
2030 PRINT CHR$(5);
2040 PRINT "[HOME][DOWN][DOWN]YOU HAVE D
ONE WELL !!!"
2050 PRINT "BUT DARE YOU TRY AGAIN AND K
EEP THE "
2060 PRINT "SCORE ROLLING"
2070 PRINT "[DOWN][DOWN][DOWN][DOWN][DOW
N][DOWN][DOWN][RVSON][RIGHT][RIGHT][RIGH
T][RIGHT]PRESS ANY KEY TO CONTINUE"
2075 FOR T=1 TO 700:NEXTT
2080 GET A$:IF A$="" THEN 2080
2090 PRINT"[CLEAR]";:ENX=9:G=2:IFG>10T
HENG=>10:PRINT"[HOME][DOWN][DOWN][DOWN]
DOWN][DOWN][DOWN][DOWN]";TAB(34);TR
2110 GOTO 270
```

# KEYCHASE BY PAUL EVANS

Fortunately, there are only two keys to master in this fast action levels-and-ladders game for the BBC micro. These are: Z — left X — right.

The object of the game is to collect the keys which you'll find scattered about the screen, then return to the starting door to move onto the next screen. You move up a level simply by running into a ladder, and down again by dropping through any of the holes in the floors.

Just to make things a little more complex, you'll find yourself pursued by Pacmen, bent on your destruction. These will move relatively slowly on the first screen, but speed up later on, whilst you find yourself slowing in comparison.

Finally a tip from the author: "Move at just the same speed as the Pacmen, because they can match any speed that you move at."

```

10 MODE2
11 GOTO1000
12 *FX11,0
20 PROCUDGS
30 VDU23:8202:0:0:0:
39 PR=1:PS=1
40 MX=967:MY=127:AX=135:AY=991:BX=839
:BY=607:CY=7:CY=511:SO=0:OS=0:S=0:PI=64:
P2=64:P3=64:MZ=130
50 GOTO2000
110 GOTO50
1000DEFFPROCUDGS
1010 VDU23,128,28,12,72,62,9,20,34,17
1020 VDU23,129,56,48,18,124,144,40,68,1
36
1030 VDU23,130,28,28,8,127,8,20,34,65
1040 VDU23,131,66,126,66,126,66,126,66,
126
1050 VDU23,132,90,255,255,255,255,255,2
53,152
1060 VDU23,133,18,63,127,63,31,63,127,5
0
1070 VDU23,134,72,252,254,252,248,252,2
54,76
1080 VDU23,135,0,0,32,80,95,85,32,0
1090 VDU23,136,8,8,24,24,8,8,8
1100 VDU23,137,60,126,252,248,252,126,6
0,0
1110 VDU23,138,60,126,63,31,63,126,60,0
1120 VDU23,139,68,238,254,254,254,124,5
6,0
1130 VDU23,140,255,255,255,255,255,255,
255,255
1140 ENDPROC
2000 VDU4:REM SCREEN 1
2010 CLS:COLURLC:PRINT"SCREEN NUMBER:
":PS
2020 COLOUR13:PRINT" "":VDU1
35:PRINT:COLOUR1:VDU133,132,132,132,134:
COLOUR0:VDU140:COLOUR1:VDU133,132,132:CO
LOUR7:VDU131:COLOUR1:VDU132,132,132,132,
132,132:COLOUR7:VDU131:COLOUR1:VDU132,13
2,134
2030 COLOUR7:PRINT" "":VDU131:P
RINT" "":VDU131:PRINT:PRINT" "":VDU131:PRINT
" "":VDU131:PRINT" "":VDU131:PRINT
2040 COLOUR1:VDU133,134:COLOUR0:VDU140:

```

```

COLOUR1:VDU133,132:COLOUR7:VDU131:COLOUR
1:VDU134:COLOUR0:VDU140:COLOUR1:VDU133,1
32,132:COLOUR7:VDU131:COLOUR1:VDU132,132
,132,132,132,132,132,134
2050 COLOUR7:PRINT" "":VDU131:PRINT
" "":VDU131:PRINT:COLOUR13:VDU135:CO
LOUR7:VDU131" "":VDU131:PRINT" "":
VDU131:PRINT
2060 COLOUR1:VDU133,132,134:COLOUR0:VDU
140:COLOUR1:VDU133,132,132,132,132,132,1
32,132,132,132,132,132,132,132:COLOUR7:V
DU131:COLOUR1:VDU134
2070 COLOUR7:PRINT" "":
VDU131:PRINT:COLOUR7:PRINT" "":VDU131:PRINT
" "":VDU131:PRINT
2080 COLOUR1:VDU133,134:COLOUR0:VDU140:
COLOUR1:VDU133,132,132,132,132,132,132:C
OLOUR7:VDU131:COLOUR1:VDU134:COLOUR0:VDU
140:COLOUR1:VDU133,132,132:COLOUR7:VDU13
1:COLOUR1:VDU132,132,134
2090 PRINT" "":COLOUR7:VDU131:
PRINT" "":VDU131:PRINT:PRINT" "":VDU13
" "":COLOUR7:VDU131:PRINT" "":VDU13
1:PRINT
2100 COLOUR1:VDU133,132,132,132,132,132,1
34:COLOUR0:VDU140:COLOUR1:VDU133:COLOUR7:V
DU131:COLOUR1:VDU132,132,132,132,132,132
,132,132,132,132,134
2110 COLOUR7:PRINT" "":VDU131:PR
INT:PRINT" "":VDU131:PRINT"
2120 COLOUR13:VDU135:COLOUR0:VDU140
2130 COLOUR1:VDU133,132,134:COLOUR0:VDU
140:COLOUR1:VDU133,132,132,132,132,132,1
32,132,132:COLOUR7:VDU131:COLOUR1:VDU132
,132,132,132,132,134
2140 PRINT" "":COLOUR7:VDU1
31:PRINT:COLOUR0:VDU140:COLOUR13:VDU135:
PRINT" "":COLOUR7:VDU131:PRINT
2150 COLOUR1:VDU133,132,132,132,132,132
:COLOUR7:VDU131:COLOUR1:VDU134:COLOUR0:V
DU140:COLOUR1:VDU133,132,132,132,132,132
,132,132:COLOUR7:VDU131:COLOUR1:VDU132,1
34
2160 PRINT" "":COLOUR7:VDU131:PRIN
T" "":VDU131:PRINT:PRINT" "":VDU
131:PRINT
2170 COLOUR1:VDU133,132,132:COLOUR7:VDU
131:COLOUR1:VDU132,132,132,132,132,134:C
OLOUR0:VDU140:COLOUR1:VDU133,132:COLOUR
7:VDU131:COLOUR1:VDU132,132,132,132,132,1
34
2180 COLOUR7:PRINT" "":VDU131:PRINT"
"":VDU131:PRINT:PRINT" "":VDU1
31:PRINT" "":COLOUR13:VDU135:COLOUR7:
PRINT" "":VDU131:PRINT
2190 COLOUR1:VDU133:COLOUR7:VDU131:COLO
UR1:VDU132,132,132,132,134:COLOUR0:VDU14
0:COLOUR1:VDU133,132,132,132,132,132,132
,132,132,132,132,134
2200 COLOUR7:PRINT" "":VDU131:PRINT:PRIN
T" "":VDU131:PRINT" "":COL
OUR11:VDU136:PRINT
2210 COLOUR1:VDU133,132,132,132,132,132,1
32,132,132,132,132,132,132,132,132,132

```

# 88C

```

,132:COLOUR7:VDU131:COLOUR1:VDU132,134
2210 PRINT"                ":COLOUR7:
VDU131:PRINT:PRINT"                ":C
OLOUR7:VDU131
2215 VDU5:MOVEMX,MY:GCOL0,5:VDU130:MOVE
AX,Y:GCOL0,3:VDU139

2220 GOSUB8000:IFSO=-1THENS0=0:PS=PS+1:
FR=FR+1:GOTO40
2225 IFPS=7ORPS=13ORPS=19ORPS=25ORPS=31
ORPS=37ORPS=43ORPS=49ORPS=55ORPS=61THENP
R=1
2230 IFPR=3ORPR=6THENGOSUB9010:GOSUB902
0:GOSUB9000:GOTO2300
2240 IFPR=2ORPR=5THENGOSUB9000:GOSUB901
0:GOTO2300
2250 GOSUB9000
2300 IFOS=-1THEN10000
2310 GOTO2220
8000 SOUND2,-10,77,1:BN=50-2*PS
8001 IFBN<1THENBN=2
8010 AS=INKEY$(BN):IFAS=""THENMOVEMX,MY
:GCOL0,5:VDUMZ:RETURN
8020 IFAS<>"Z"THEN8030
8025 IFMX>64THENMX=MX-64:GOTO8050
8027 RETURN
8030 IFAS<>"X"THEN8040
8035 IFMX<1215THENMX=MX+64:GOTO8050
8040 RETURN
9050 IFPOINT(MX+32,MY-16)<>11THEN8060
8055 IFS=5THENS0=0:GOTO8600
8057 GOTO8500
8060 IFPOINT(MX+32,MY-20)<>7THEN8070
8062 IFAS="Z"THENMOVEMX+64,MY:GCOL0,0:V
DU140
8063 IFAS="X"THENMOVEMX-64,MY:GCOL0,0:V
DU140
8065 FORI=1TO3:MY=MY-32:MOVEMX,MY:GCOL0
,5:VDU130:MOVEMX,MY-32:GCOL0,0:VDU130:MO
VEMX,MY-32:GCOL0,7:VDU131:SOUND2,-10,97+
I*40,1:NEXTI:IFPOINT(MX+32,MY-20)<>7THEN
8020
8066 GOTO8060
8070 IFPOINT(MX+32,MY-44)<>0THEN8080
8072 IFAS="Z"THENMOVEMX+64,MY:GCOL0,0:V
DU140
8073 IFAS="X"THENMOVEMX-64,MY:GCOL0,0:V
DU140
8075 FORI=1TO3:MY=MY-32:MOVEMX,MY:GCOL0
,5:VDU130:MOVEMX,MY-32:GCOL0,0:VDU130:SO
UND2,-10,250-I*40,1:NEXTI:GOTO8020
8080 IFPOINT(MX+32,MY-16)=13THENS=S+1:M
OVEMX,MY:GCOL0,0:VDU140:SOUND3,-10,253,2
8090 IFPOINT(MX+32,MY-16)=3THENS=-1:RE
TURN
8399 SOUND1,-10,121,1
8400 IFAS="Z"THENMOVEMX,MY:GCOL0,5:VDU1
28:MOVEMX+64,MY:GCOL0,0:VDU140:MZ=128:RE
TURN
8410 IFAS="X"THENMOVEMX,MY:GCOL0,5:VDU1
29:MOVEMX-64,MY:GCOL0,0:VDU140:MZ=129:RE
TURN
8500 IFAS="Z"THENMX=MX+64:RETURN
8510 IFAS="X"THENMX=MX-64:RETURN
8600 REM CHANGE SCREEN
8650 SO=-1:RETURN
9000 MO=P1:X=AX,Y=AY:PROCCHASE:P1=MO:AX
=X:AY=Y:RETURN

```

```

9010 MO=P2:X=BX,Y=BY:PROCCHASE:P2=MO:BX
=X:BY=Y:RETURN
9020 MO=P3:X=CX,Y=CY:PROCCHASE:P3=MO:CX
=X:CY=Y:RETURN
9100 DEFPROCCHASE
9110 IFX>MO>1279THEN9500
9120 IFX>MO<0THEN9500
9125 SD=INT(RND(1)*3):IFSD=1THEN9130
9126 GOTO9140
9130 IFX=MX THEN MO=-64
9135 IFX=MX THEN MO=64
9140 IFPOINT(X+MO+32,Y-16)=11THEN9500
9150 IFPOINT(X+MO+32,Y-16)=13THEN9500
9160 IFPOINT(X+MO+32,Y-16)=3ORPOINT(X+M
O+32,Y-12)=5THENS=-1:ENDPROC
9165 IFX=MX AND Y=MY THEN OS=-1:ENDPROC

9170 PP=0:IFPOINT(X+MO+32,Y-20)<>7THEN9
180
9172 MOVEX,Y:GCOL0,0:VDU140
9175 FORI=1TO3:Y=Y+32:MOVEMO+X,Y:GCOL0,
3:VDU139:MOVEMO+X,Y-32:GCOL0,0:VDU139:MO
VEMO+X,Y-32:GCOL0,7:VDU131:SOUND2,-10,97
+I*40,1:NEXTI:X=X+MO:PP=1
9176 MOVEMX,MY:GCOL0,5:VDUMZ
9177
9180 IFPOINT(MO+X+32,Y-44)<>0THEN9190
9182 MOVEX,Y:GCOL0,0:VDU140
9185 FORI=1TO3:Y=Y-32:MOVEMO+X,Y:GCOL0,
3:VDU139:MOVEMO+X,Y-32:GCOL0,0:VDU139:SO
UND2,-10,250-I*40,1:NEXTI:X=X+MO:PP=1
9186 MOVEMX,MY:GCOL0,5:VDUMZ
9190 REM
9400 IFMO=-64THENX=X+MO:MOVEX,Y:GCOL0,3
:VDU138:MOVEX+64,Y:GCOL0,0:VDU140:GOTO99
98
9410 IFMO=64THENX=X+MO:MOVEX,Y:GCOL0,3:
VDU137:MOVEX-64,Y:GCOL0,0:VDU140:GOTO999
8
9500 MO=-MO
9998 REM
9999 ENDPROC
10000 MODE7:SOUND1,-10,5,15:PRINTTAB(10,
10)::VDU141:PRINT"YOU'RE DEAD!!!"
10001 VDU141:PRINT"                YOU'RE D
EAD!!!"
10010 PRINT:VDU141:PRINT" YOU SCORED:"
PS*5+S
10020 VDU141:PRINT" YOU SCORED:"PS*5+S
10999 END
11000 CLS:INPUT"COLOUR OF PAC-MAN: "PMC
11010 INPUT"COLOUR OF YOU: "MC
11011 PRINT
11020 INPUT"KEY COLOUR: "KC
11021 PRINT
11030 INPUT"DOOR COLOUR: "DC
11031 PRINT
11040 INPUT"FLOR COLOUR: "FC
11041 PRINT
11050 INPUT"LADDER COLOUR: "LC
11051 PRINT
11055 INPUT"BACKGROUND COLOUR: "BC
11060 VDU19,3,PMC,0,0,0:VDU19,5,MC,0,0,0
:VDU19,13,KC,0,0,0:VDU19,11,DC,0,0,0:VDU
19,1,FC,0,0,0:VDU19,0,BC,0,0,0:VDU19,7,L
C,0,0,0
11070 GOTO15

```

# BREAKOUT BY MARK GANNON

This fast-action Vic-20 version of the classic game *Breakout* certainly deserves a look.

Use the keys:

← Left → Right

to move the bat. Hit the bouncing ball to destroy the rows of coloured bricks. The ball has an annoying habit of changing direction when it knocks a brick out of the wall, so you're never quite sure where it'll return.

The program loads in two sections which you'll have to save, one

after the other on tape. When typing in the listing, you'll often find words within square brackets, indicating the keys you should press. This method eliminates all those nasty-looking graphics characters.

So:

[HOME]

— Press the HOME key.

[RED]

— Colour Red.

[X]

— Press key X whilst pressing SHIFT.

[C]

— Press key X whilst pressing C.

```

10 POKE36879,8
20 PRINT"[CLEAR][WHITE][RIGHT][RIGHT][RI
GHT][RIGHT][RIGHT][RIGHT][RIGHT][RVSON]B
REAKOUT[RVSOFF]"
30 PRINT"[DOWN][CYAN]THE MOVEMENT KEYS A
RE:"
40 PRINT"[DOWN][DOWN][RIGHT][RIGHT][RIGH
T][RIGHT][RIGHT][YELLOW]LEFT". "[RIGHT]RI
GHT"
50 PRINT"[DOWN][WHITE][RIGHT][RIGHT][RIG
HT][RIGHT][RIGHT][RIGHT][RIGHT]. "[RIGH
T][RIGHT][RIGHT]/"
60 PRINT"[DOWN][DOWN][GREEN]THE LEFT SID
E OF THE [DOWN]BAT HITS "
70 PRINT"THE BALL TO [DOWN]THE LEFT, TH
E RIGHT TO[DOWN]THE RIGHT."
80 FOR I=1 TO 2000: NEXT: PRINT"[DOWN][DOWN][
WHITE][RIGHT][RIGHT][RIGHT][RIGHT]THE GAME IS N
OW[RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RI
GHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT]L
OADING"
90 FOR I=1 TO 2000: NEXT: PRINT"[BLACK][DOWN]
[DOWN][DOWN][DOWN][DOWN][DOWN][DOWN][DOW
N][DOWN][DOWN]"
100 POKE198,5: POKE631,78: POKE632,69: POKE
633,87: POKE634,13: POKE635,131: END
    
```

```

10 S1=36875: V=36878: S2=36876: SC=0: BA=1: G
OTO60
20 UD=-UD: SC=SC+(PEEK(30720+W)AND15)*10:
30 IF INT(SC/5600)*5600=SC THEN GOSUB420: IF
BA>1 THEN BA=BA-1
40 REM**BLIP WHEN BALL HITS BRICK**
50 POKES1=1,240: POKES1=1,0: RETURN
55 REM**MAIN LOOP**
60 AS="[DOWN][DOWN][DOWN][DOWN][DOWN][DO
WN][DOWN][DOWN][DOWN][DOWN][DOWN][DOWN]
[DOWN][DOWN][DOWN][DOWN][DOWN][DOWN][
DOWN][DOWN]": BS="[RIGHT][RIGHT][RIGHT][
RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT]
[RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RI
GHT][RIGHT]"
70 DEFNZ(A)=INT(RND(1)*A): POKEV,15
80 CS="[WHITE][RVSON][RVSOFF]
[RVSON][RVSOFF]"
85 FS="[WHITE][RVSON][BLACK]
[WHITE][RVSOFF]"
90 GOSUB340
100 GOSUB420
110 P=11: Q=1: V=10: LR=1: UD=1
120 VI=V+(PEEK(197)=29)-(PEEK(197)=30): I
F VI<1 OR VI>19 THEN VI=V
    
```

# UNEXPANDED VIC 20

```

130 PRINT"[HOME]";A$:TAB(V);"[BLACK][RVS
ON] [RVSOFF][HOME]";A$:TAB(VI);"[WHITE]
[c T][c T]";V=VI
135 IFPEEK(653)AND1THEN135
140 G=P+UD:IFG=2ORG=20THENUD=-UD:GOSUB48
0
150 F=Q+LR:IFF=1ORF=20THENLR=-LR
160 IFG=20ANDF<>1ANDF<>20THENLR=SGN(F-V-
.5)
170 IFG=11ANDFNZ(3)-1ANDF<18ANDUD=1THENF
=Q
180 IFPEEK(W+22)=160THEN230
190 W=G*22+F+7680:IFPEEK(W)=208THENGOSUB
200
200 PRINT"[HOME]";LEFT$(A$,P);LEFT$(B$,Q
)" "
210 PRINT"[HOME]";LEFT$(A$,G);LEFT$(B$,F
)"[YELLOW]";[s Q]";Q=F:P=G
220 GOTO120
230 W=0:IFSC>HITHENHI=SC
235 IFBA=3THEN280
237 REM**MISSED BALL**
240 PRINT"[HOME]"A$"[UP][RVSON][WHITE]
MISSED IT [RVSOFF]";
250 BA=BA+1:PRINT"[HOME][GREEN]"LEFT$(B$
.20)BA
260 FORI=1TO1000:NEXT:PRINT"[HOME]"A$"[U
P][UP]"C$C$F$;
270 GOTO110
275 REM**END OF GAME**
280 PRINT"[HOME][DOWN][DOWN]"LEFT$(A$,10
)"[RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RI
GHT][WHITE]GAME OVER"
290 PRINT"[DOWN][DOWN][RIGHT][RIGHT][RIG
HT][RIGHT]PRESS ANY KEY"
300 FORI=1TO1000:NEXT
305 GOSUB490
320 GOTO10
330 FORI=1TO1000:NEXT:GOTO10
335 REM**TITLE PAGE**
340 POKE36879.25:PRINT"[CLEAR]";PRINT:PR
INT:PRINT"[PURPLE] [RVSON] BREAKOUT
[RVSOFF]"
350 PRINT:PRINT"[RIGHT][RIGHT][RIGHT][BL
ACK]BY MARK GANNON"
360 PRINT:PRINT:PRINT:PRINT"[RED] LEF
T","RIGHT";PRINT:PRINT"[BLACK]
"/"
365 PRINT:PRINT:PRINT"[RIGHT][RIGHT][RIG
HT][RIGHT][BLUE]HI SCORE:"HI
366 PRINT:PRINT"[BLACK][RIGHT][RIGHT]SHI
FT-LOCK:PAUSE"
370 PRINT:PRINT"[PURPLE] [s o][c T][c T
][c T][c T][c T][c T][c T][c T][c T][c T]
[c T][c T][c T][c T][s P]";PRINT"[PURPL
E] [c G][BLACK]PRESS ANY KEY.[PURPLE][c
M]"

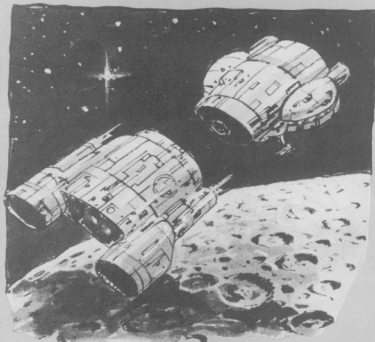
```

```

380 PRINT" [s L][c o][c o][c o][c o][c o]
[c o][c o][c o][c o][c o][c o][c o][c o]
[c o][c o]"
390 FORI=1TO1000:NEXT
395 POKE198,0
400 GETTS:IFTS=-1THEN400
410 RETURN
415 REM*SCREEN SETUP*
420 POKE36879.8:PRINT"[CLEAR][DOWN][WHIT
E][RVSON] [RVSOFF]"
430 FORI=1TO20:PRINTC$;NEXT
440 PRINT"[HOME][DOWN][DOWN][DOWN][DOWN]
";
450 FORI=7TO1STEP-1:PRINTMID$("[WHITE][R
ED][CYAN][PURPLE][GREEN][BLUE][YELLOW]",
I,1)"[RVSON][RIGHT][s P][s P][s P][s P][
s P][s P][s P][s P][s P][s P][s P][s P][
s P][s P][s P][s P][s P][s P][s P][s P][
RIGHT][RVSOFF]";NEXT
460 PRINT"[HOME][GREEN]SCORE"SC"[HOME][R
IGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT]
[RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGH
T][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT]BA
465 PRINT"[DOWN][DOWN][DOWN][DOWN][DOWN]
[DOWN][DOWN][DOWN][DOWN][DOWN][DOWN][DOW
N][DOWN][DOWN][DOWN][DOWN][DOWN][DOWN][D
OWN][RVSON][BLACK][RIGHT]
[RVSOFF]";
470 RETURN
475 REM**BLIP WHEN BALL HITS BAT**
480 POKES1,240:POKES1,0:RETURN
485 REM**MUSIC LOOP**
490 POKE198,0
495 READN:IFN=-1THENRESTORE:GOTO490
500 READM
510 POKES1,N:POKES2,N
520 FORI=1TOM:NEXTI
530 POKES1,0:POKES2,0
540 FORI=1TO10:NEXTI
550 GETTS:IFTS=-1THEN495
555 RESTORE:RETURN
560 DATA201,240,201,240,201,120,191,120,
215,120,201,120,191,120,175,120,0,5
565 DATA195,240,195,240,195,120,183,120
570 DATA191,120,195,120,191,120,183,120,
0,5,201,240,201,240,201,120,191,120
580 DATA215,120,201,120,191,120,175,120,
159,120,175,120,195,120,191,120,195,120,
183,120
590 DATA175,120,201,120,201,120,191,120,
195,120,201,120,191,120,195,120,201,120,
191,120
600 DATA215,120,201,120,195,120,175,120,
195,120,183,120,191,120,195,120,183,120,
191,120
700 DATA195,120,183,120,191,120,195,120,
191,120,183,120,201,120,191,120,195,120,
201,120
705 DATA191,120,195,120
710 DATA201,120,191,120,215,120,201,120,
191,120,175,120,159,120,195,120,191,120,
183,120
720 DATA175,120,167,120,175,240,0,490,-1

```

# SPECTRAUT BY NIBBLESOFT



The deep space shuttle mission started out peacefully enough — but then trouble. Your fuel tanks are empty.

Your only hope is to crash land on a nearby planet and collect fuel in the hope of being able to resume your mission.

But beware the flying saucers and deadly laser which could mean total destruction.

Full instructions for Spectraut are included in the program.

```
10 MODE 1:INK 0,0:BORDER 0:INK 1,24:INK
2,26:INK 3,2
20 SYMBOL AFTER 97:SYMBOL 97,31,127,255,
255,254,254,255,127:SYMBOL 99,63,31,7,1,
0,0,0,0:SYMBOL 100,192,224,240,248,252,1
26,126,63:SYMBOL 101,0,0,0,0,127,255,255
,127:SYMBOL 102,63,63,127,255,255,254,25
2,248
30 PAPER 3:PEN 1:PRINT"
```

ad  
cd  
effect

```
RAUTAUT "PAPER 1:PEN 3:P
RINT" BY NIBBLESOFT
```

```
40 PAPER 0:PEN 2:PRINT"AS YOU WERE FLYIN
G THROUGH SPACE IN YOURSPACE SHUTTLE YOU
RAN OUT OF FUEL AND HAD TO CRASH LAND
ON AN ALIEN PLANET. YOU NOW HAVE TO R
E-FUEL YOUR SHUTTLE TO TAKE OFF SAFEL
Y."
```

```
50 PRINT:PRINT"YOU DO THIS BY FLYING UP
TO THE OIL DRUM AT THE TOP OF THE SCREEN
AND RETURNING TO YOUR SHUTTLE AT THE BO
TTOM LEFT OF THE SCREEN."
```

```
60 PRINT:PRINT"YOUR SHUTTLE NEEDS 10 UNI
TS OF FUEL TO FLY, SO YOU WILL HAVE TO
MAKE THE JOURNEY TEN TIMES."
```

```
70 PRINT:PRINT"HIT ENTER TO CONTINUE"
80 RESTORE 120:ENV 5,15:-1,10:ENV 6,15,-
1,2
```

```
90 READ N%,D%,E%:IF N%=-1 THEN 80
100 SOUND 1:N%*2,D%*10,15,E%*5
```

```
110 IF INKEY(18)=0 THEN 200 ELSE 90
120 DATA 1517,4,1,1517,4,1,1517,4,1,1517
,4,1,1517,4,1,1517,4,1,1517,4,1
125 DATA 478,2,0,506,2,0,568,4,0,478,2,0
,506,2,0,568,4,0,478,2,0,506,2,0,568,2,0
,632,2,0,568,2,0,506,2,0
130 DATA 478,2,0,506,2,0,568,4,0,478,2,0
,506,2,0,568,4,0,478,2,0,506,2,0,568,2,0
,632,2,0,568,2,0,506,2,0
140 DATA 426,2,0,478,2,0,506,4,0,426,2,0
,478,2,0,506,4,0,426,2,0,478,2,0,506,2,0
,568,2,0,506,2,0,478,2,0
150 DATA 426,2,0,478,2,0,506,4,0,426,2,0
,478,2,0,506,4,0,426,2,0,478,2,0,506,2,0
,568,2,0,506,2,0,478,2,0
160 DATA 478,2,0,506,2,0,568,4,0,478,2,0
,506,2,0,568,4,0,478,2,0,506,2,0,568,2,0
,632,2,0,568,2,0,506,2,0
170 DATA 478,2,0,506,2,0,568,4,0,478,2,0
,506,2,0,568,4,0,478,2,0,506,2,0,568,2,0
,632,2,0,568,4,0,379,8,0
180 DATA -1,-1,-1
200 CLS:PRINT"WHEN YOU GET FUEL FROM THE
OIL DRUM THE COMPUTER WILL BLEEP. YOU W
ILL HERE THIS SOUND AGAIN WHEN YOU TAKE
IT BACK TO THE SHUTTLE."
```

```
210 PRINT:PRINT"THE AMOUNT OF FUEL IN TH
E SHUTTLE IS SHOWN BY A REN LINE ON I
T."
```

```
220 PRINT:PRINT"Life is not that simple
though, on your journey you must watch o
ut for the flying saucers and the d
```



## 13

# SPECTRA NUT

## AMSTRAD CPC 464

```
650 SYMBOL 97,255,195,219,219,195,255,19
5,231:SYMBOL 98,231,195,255,223,223,223,
195,255
```

```
660 PEN 1:PRINT"*****":F
OR I=1 TO 23:PRINT"*" *
NEXT PEN 2:PRINT"*****"
```

```
670 LOCATE 1,1:PRINT CHR$(22)+CHR$(1):PE
N 8:LOCATE 1,22:PRINT"n":PRINT"d":PRINT"
9":LOCATE 1,1:PRINT CHR$(22)+CHR$(0)
```

```
680 PEN 8:PRINT CHR$(22)+CHR$(1):LOCATE
1,13:PRINT CHR$(142):"
```

":CHR\$(143):PRINT CHR\$(22)+CHR\$(0)

```
690 PEN 10:LOCATE 20,1:PRINT"a":LOCATE 2
0,2:PRINT"b"
```

```
700 FOR I=39200 TO 39207:POKE 1,0:NEXT:P
OKE 39204,4:POKE 39202,100:POKE 39206,2:
```

```
POKE 39207,1:FOR I=39210 TO 39217:POKE 1
,0:NEXT:POKE 39214,72:POKE 39212,100:POK
E 39216,1:POKE 39217,2
```

```
710 FOR I=39220 TO 39227:POKE 1,0:NEXT:P
OKE 39224,72:POKE 39222,183:POKE 39226,1
```

```
:POKE 39227,2:FOR I=39230 TO 39237:POKE
1,0:NEXT:POKE 39234,4:POKE 39232,31:POKE
39236,2:POKE 39237,1
```

```
720 FOR I=41000 TO 41063:POKE 1,0:NEXT
730 REM*****
```

```
***
740 FLAG%=0:FUEL%=0:LIVES=3
```

```
750 X%=52:Y%=25:DIR%=64
```

```
760 SPRITE=39000:ALIEN=39000:DETECT=3700
0
```

```
770 IF INKEY(71)=0 AND X%>6 THEN CALL SP
RITE,X%,Y%,1000:DIR%=64:X%=X%-2
```

```
780 IF INKEY(63)=0 AND X%<70 THEN CALL S
PRITE,X%,Y%,1000:DIR%=128:X%=X%+2
```

```
790 IF INKEY(47)=0 AND Y%<181 THEN CALL
SPRITE,X%,Y%,1000:Y%=Y%+2:SOUND 1,0,5,12
```

```
,0,0,5 ELSE IF Y%>25 THEN Y%=Y%-2
```

```
800 IF RND>0.97 THEN FX=1:PLOT 0,200,9:0
RAW 639,200:SOUND 2,50,0,15,2,1
```

```
810 CALL SPRITE,X%,Y%,DIR%
```

```
820 IF X%=70 AND Y%=179 THEN GOSUB 880
```

```
830 IF X%=6 AND Y%=25 THEN 900
```

```
840 CALL ALIEN,39200:CALL ALIEN,39210:CA
LL ALIEN,39220:CALL ALIEN,39230
```

```
850 CALL DETECT,X%,Y%:IF PEEK(37200)=1 T
HEN 990
```

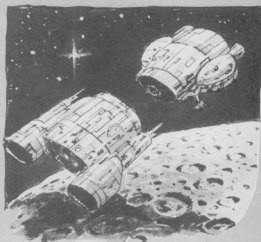
```
860 IF FX=1 THEN PLOT 0,200,0:DRAW 639,2
00:FX=0
```

```
870 GOTO 770
```

```
880 IF FLAG%=0 THEN FLAG%=1:FUEL%=FUEL%+
1:SOUND 2,50,5,15
```

```
890 RETURN
```

```
900 IF FLAG%=0 THEN 840
```



```
910 FLAG%=0:PLOT 0,16:FUEL%+2,3:DRAW 12,
16:FUEL%+2:SOUND 2,50,5,15:IF FUEL%<10 T
HEN 840
```

```
920 CLS:SYMBOL AFTER 42:SYMBOL 104,0,0,1
,1,3,3,7,7:SYMBOL 98,0,0,0,15,19,35,35,6
```

```
7:SYMBOL 99,0,0,0,0,255,255,17,119,17:SYMB
OL 100,15,15,31,255,254,126,126,126:SYMB
```

```
OL 101,131,255,255,255,254,255,255,127:S
YMBOL 102,221,17,255,252,0,255,255,255
```

```
,254:FOR I=1 TO 50:PLOT RND*640,RND*400,
RND*15:NEXT
```

```
940 WINDOW R1,9,11,1,25:LOCATE R1,1,25:P
EN R1,8:PRINTR1," hbcdefg"
```

```
950 FOR I=25 TO 1 STEP-1:PRINT R1," " :SOU
ND 1,1*10,2,15:FOR D=1 TO 40:NEXT:NEXT
```

```
960 CLS:PRINT"HELL DONE YOU FILLED YOUR S
HUTTLE WITH FUEL AND TOOK OFF OK"
```

```
970 PRINT:PRINT"HIT ENTER TO PLAY AGAIN"
```

```
980 IF INKEY(18)=0 THEN RUN ELSE 980
```

```
990 SOUND 1,2000,0,15,1,0,15
```

```
1000 FOR I=1 TO 50:INK 7,6:FOR D=1 TO 20
:NEXT:INK 7,25:FOR D=1 TO 20:NEXT:NEXT I
```

```
1010 CALL SPRITE,X%,Y%,1000:LIVES=LIVES-
1:IF LIVES=0 THEN 1030
```

```
1020 FOR D=1 TO 200:NEXT:GOTO 750
```

```
1030 CLS:PRINT "GAME OVER":PRINT:PRINT:P
RINT"HIT ENTER"
```

```
1040 IF INKEY(18)=0 THEN RUN ELSE 1040
```

# RED FRED BY AIDEN TAYLOR

## SPECTRUM 16/48K

This is a very colourful version of the classic game Q\*bert, with all the original features such as the bouncing ball, the snake and the saucers.

The idea of the game is that you must use the keys:

Q — Up Left P — Up Right

A — Down Left <ENTER> — Down Right

to move Fred about the pyramid, painting the blocks as you go. As in all the best games, any other moving object is dangerous to Fred's health, so avoid the balls and snake. If things get a bit too dicey, just jump onto one of the magenta saucers and you'll be immediately transported to the top of the pyramid — probably closely followed by the snake. The snake becomes progressively more intelligent as you complete each sheet, so be careful.

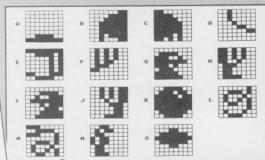
When typing the listing, you'll come across underlined characters. These are in place of the user-defined graphics which are normally scattered through the listing, and should be typed in GRAPHICS mode (the G cursor). So if you see the character E, then enter graphics mode (CAPS SHIFT + 9), then hit the character I, then return to normal mode (CAPS SHIFT + 9 again). You may also come across a set of characters such as sh8. These refer to Spectrum symbol graphics (above the keys 1 to 8). Again in graphics mode, hold down the SHIFT key whilst pressing the number (in this case 8). When the program has been run, a listing will show the graphics, shown in the diagram below.

Finally, if anybody wants to make changes to the program, here are a couple of tips. To get rid of the sound, change line 115 to:

115 FOR z=1 to 10: NEXT z: LET 1=RND

People with black-and-white televisions may find that the following helps:

340 IF c=6 THEN LET c=4



```

122 IF l<f THEN GO TO 130
125 IF u<y THEN IF a<x THEN
LET u=u+(u<y)*2-(u>y)*2: LET a=
a+(a<x)*2-(a>x)*2: GO TO 130
126 IF u<y THEN IF a=x THEN
LET o=RND: LET a=a+(o*.5)*2-(o<.
5)*2: LET u=u+(u<y)*2-(u>y)*2: G
O TO 130
127 IF u=y THEN IF a<x THEN
LET o=RND: IF u<17 THEN LET u=
u+(o*.5)*2-(o<.5)*2: LET a=a+(a<x
)*2-(a>x)*2
130 LET h=h+(1+.5)*2-(1<.5)*2:
LET v=v+2: IF v>17 THEN LET v=3
: LET h=15

```



# RED FRED BY AIDEN TAYLOR



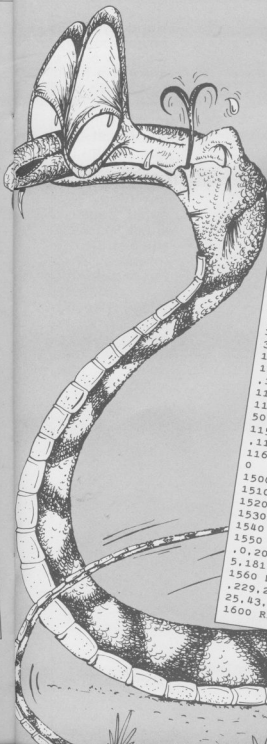
```

145 IF ATTR (Y,X-1)=7 THEN PRI
    T INK C:AT Y,X-1;"Bsh8C":AT Y+
    1,X-1;"DEF":AT Y-1,X;"A": LET T=
    T+1: LET S=S+10: PRINT INK 6:AT
    0.6:S: GO TO 160
150 PRINT INK (ATTR (Y,X-1)):A
    T Y-1,X;"A":AT Y,X;"sh8"
160 IF INKEY$=CHR$ 13 THEN LET
    X=X+2: LET Y=Y+2: LET P=P+2
162 IF INKEY$="p" THEN LET X=X
    +2: LET Y=Y-2: LET P=P+2
165 IF INKEY$="q" THEN LET X=X
    -2: LET Y=Y-2: LET P=P-1
167 IF INKEY$="a" THEN LET X=X
    -2: LET Y=Y+2: LET P=P-1
170 GO TO 100
200 IF ATTR (Y+1,X)=3 THEN GO
    TO 600
205 REM YOUR DEAD !
210 PAUSE 10: BEEP .4,9: BEEP .
    4,0: BEEP .2,4: BEEP .2,5
215 FOR F=0 TO 100: NEXT F
220 PRINT INK (ATTR (U,A-1)):A
    T U-1,A;"A":AT U,A;"sh8": INK (A
    TTR (V,H-1)):AT V,H;"sh8"
230 LET M=M-1: PRINT INK 6:AT
    0.22:"LIVES:"M$( TO M):" "
235 IF ATTR (Y,X)=71 THEN PRIN
    T INK 7:AT Y-1,X:" ":AT Y,X:" "
240 IF M<1 THEN GO TO 400
    
```

```

250 LET U=17: LET A=(INT (RND*8
    )+4)+1: LET Y=11: LET X=15: LET
    P=1: LET H=15: LET V=3
260 GO TO 100
300 REM NEW SHEET
305 PRINT INK 6:AT 0.22:"BONUS
    ":b: LET Z=USR 30025
310 FOR F=1 TO B STEP 10: LET S
    =S+10: PRINT INK 6:AT 0.6:S:AT
    0.28;b-F:" ": NEXT F
320 IF R>.2 THEN LET R=R-.1
340 LET B=B+50: LET T=0: LET C=
    C+1: IF C=7 THEN LET C=4
350 CLS : GO TO 20
400 REM GAME OVER
410 PRINT AT 1,11: PAPER 7: INK
    2: FLASH 1:"GAME OVER"
420 FOR F=0 TO 1: PRINT BRIGHT
    F: INK 5:AT 21.6:"PRESS ANY KEY
    TO PLAY": PAUSE 2: NEXT F
430 IF INKEY$<>" " THEN CLS : G
    O TO 10
440 GO TO 420
600 LET E=0: PRINT AT Y+1,X-1:"
    "
610 FOR F=1 TO 12
615 IF E=0 THEN PRINT INK 3:A
    T U-1,A;"M": PAPER (ATTR (U,A-1
    )):AT U,A;"L": INK 1: PAPER (ATTR
    (V,H-1)):AT V,H;"K"
620 IF P=2 THEN PRINT INK 2:A
    T Y-1,X;"I":AT Y,X;"J": GO TO 63
    0
625 PRINT INK 2:AT Y-1,X;"G":A
    T Y,X;"H"
630 PRINT INK 3:AT Y+1,X;"Q"
635 BEEP .05,F: LET L=(RND)+.1
640 IF E=0 THEN PRINT INK (AT
    TR (U,A-1)):AT U-1,A;"A":AT U,A:
    "sh8"
645 IF ATTR (U+1,A)<4 THEN LET
    E=1
650 IF L>R THEN IF U<>Y THEN
    IF A<>X THEN LET U=U+(U<Y)*2-(U
    >Y)*2: LET A=A+(A<X)*2-(A>X)*2
655 PRINT AT Y-1,X:" ":AT Y,X:"
    ":AT Y+1,X;"Q"
660 IF P=2 THEN LET X=X-1
665 IF P=1 THEN LET X=X+1
670 LET Y=Y-1
680 NEXT F
    
```

# SPECTRUM 16/48K



```
690 LET y=3: LET x=15
695 PRINT INK 2;AT y-1,x;"@";
PAPER (ATTR (y,x-1));AT y,x;"H"
700 IF e=1 THEN LET s=s+200: B
EEP .4,20: BEEP .2,20: BEEP .2,2
0: LET u=17: LET a=(INT (RND*8)*
4)+1
720 PRINT AT y,x: INK (ATTR (y,
x-1));"sh8";AT y-1,x;"A"
730 GO TO 100
1000 REM USER DEFINED GRAPHICS
1010 FOR n=144 TO 158: FOR f=0 T
O 7: READ a: POKE USR CHR$ n+f,a
: NEXT f: NEXT n
1020 DATA 0,0,0,0,0,0,60,255
1030 DATA 3,15,31,63,63,47,35
1040 DATA 192,240,248,252,252,25
2,244,212
1050 DATA 32,32,32,16,12,3,0,0
1060 DATA 255,61,5,5,5,5,199,60
1070 DATA 84,84,84,88,112,192,0,
0
1080 DATA 0,120,228,254,232,112,
248,252
1090 DATA 212,212,212,200,112,96
,96,120
1100 DATA 0,30,39,127,23,14,31,6
3
1110 DATA 43,43,43,19,14,6,6,30
1120 DATA 60,118,251,255,255,255
,126,60
1130 DATA 2,58,68,74,50,2,68,56
1140 DATA 224,60,226,2,58,68,74,
50
1150 DATA 48,88,96,176,176,96,96
,112
1160 DATA 0,24,126,255,126,24,0,
0
1500 REM M/C SOUND
1510 FOR f=30000 TO 30045
1520 READ a
1530 POKE f,a
1540 NEXT f
1550 DATA 6,1,197,33,224,3,17,40
,0,205,181,3,33,238,5,17,20,0,20
5,181,3,193,16,234,201
1560 DATA 33,244,1,17,10,0,6,255
,229,213,197,205,181,3,193,209,2
25,43,16,244,201
1600 RETURN
```

# POWERBOAT BY S A SMITH

```

10 GRAPHICS 1+16:POSITION 5,10: ? #6:"POWERBOAT":POSITION 4,12: ? #6:"BY S A SMITH"
11 POSITION 6,13: ? #6: "(C) 1984"
15 RESTORE
20 DATA 0,0,0,0,0,0,0,0
30 DATA 255,255,255,255,255,255,255,255
40 DATA 0,0,24,60,126,126,60,0
50 DATA 24,60,126,126,60,126,126,60
60 DATA 0,68,238,68,0,68,238,68
70 FOR J=0 TO 511:POKE 10240+J,PEEK(57344+J):NEXT J
80 FOR J=0 TO 39:READ JK:POKE 10240+J,JK:NEXT J
90 ? " "
99 GRAPHICS 1+16
100 ? #6:"THE AIM OF THIS GAME IS TO COLLECT THE YELLOW BUOYS BY COLLIDING
YOUR BOAT WITH THEM"
110 ? #6:"AVOID SHALLOWS MARKED WITH X'S AND AVOID RED BUOYS. YOU CONTROL THE
E BOAT WITH JOYSTICK "
120 ? #6
130 ? #6:"PRESS TRIG TO START"
140 IF STRIG(0)=1 THEN 140
150 GRAPHICS 1+16:POKE 756,40
160 POSITION 0,0: ? #6:"HIGH SCORE=":HIGH
165 SC=0
170 POSITION 0,1: ? #6:"SCORE=":SC
180 COLOR 36:FOR J=0 TO 19:PLOT J,2:PLOT J,22:NEXT J
190 JK=INT(RND(1)*3)+8
200 JK2=INT(RND(1)*3)+14
210 JK3=INT(RND(1)*3)+6
220 FOR J=JK1 TO JK2
230 PLOT JK3+INT(RND(1)*3)-1,J:DRAWTO JK3+INT(RND(1)*3)+6,J:NEXT J
240 COLOR 2:FOR J=1 TO 10:PLOT INT(RND(1)*20),INT(RND(1)*18)+3:NEXT J
250 COLOR 34:FOR J=1 TO 20:PLOT INT(RND(1)*20),INT(RND(1)*18)+3:NEXT J
260 J=INT(RND(1)*10)+6:COLOR 131:PLOT J,21:COLOR 32:PLOT J,20:PLOT J,19
270 X=J+Y-21:K=J+Y-21:EC=0:DIR=1
280 GOSUB 2000
290 TIME=0:PAUSE=50:UP=80
300 GOSUB 600
310 GOSUB 1000
320 GOSUB (DIR+100)+2900
330 GOSUB 500
340 TIME=TIME+1:IF TIME=UP THEN GOSUB 800
350 FOR JK=1 TO PAUSE:NEXT JK
400 GOTO 300
500 POSITION 6,1: ? #6:SC:RETURN
600 S=STICK(0)
610 IF S=15 THEN RETURN
620 IF S=11 AND DIR<4 THEN DIR=4:GOTO 2060
630 IF S=7 AND DIR>2 THEN DIR=2:GOTO 2020
640 IF S=14 AND DIR<1 THEN DIR=1:GOTO 2000
650 IF S=13 AND DIR>3 THEN DIR=3:GOTO 2040
660 RETURN
670 PAUSE=PAUSE-S:IF PAUSE<0 THEN PAUSE=0
680 COLOR 2:FOR J=1 TO 10:PLOT INT(RND(1)*20),INT(RND(1)*18)+3:FOR QW=240 TO 0 S
STEP -20:BOUND 1,QW,10,8:NEXT QW:NEXT J
690 COLOR 34:FOR J=1 TO 10:PLOT INT(RND(1)*20),INT(RND(1)*18)+3:FOR QW=240 TO 0 S
STEP -20:BOUND 1,QW,10,8:NEXT QW:NEXT J
700 SOUND 1,0,0,0
840 IF X>3 AND X<16 AND Y<18 AND Y>5 THEN 860
850 RETURN
860 COLOR 34:PLOT X-3,Y:PLOT X+3,Y:PLOT X,Y-3:PLOT X,Y+3
862 FOR QW=0 TO 240 STEP 30:FOR Q=1 TO 10:NEXT Q:BOUND 1,QW,10,8:NEXT QW:BOUND 1
,0,0,0:RETURN
1000 IF BOUY=1 THEN BOUY=0:GOTO 1100
1010 BOUY=1
1020 POKE 10257,24:POKE 10258,60:POKE 10259,126:POKE 10260,126:POKE 10261,60:POK
E 10262,24
1030 RETURN
1100 POKE 10257,0:POKE 10258,24:POKE 10259,60:POKE 10260,126:POKE 10261,126:POKE
10262,60
1110 RETURN
2000 POKE 10264,24:POKE 10265,60:POKE 10266,102:POKE 10267,126:POKE 10268,126:PO
KE 10269,126:POKE 10270,126
2010 POKE 10271,60:RETURN
2020 POKE 10264,0:POKE 10265,0:POKE 10266,4:POKE 10267,6:POKE 10268,255:POKE 102
69,254:POKE 10270,124
2030 POKE 10271,0:RETURN
2040 POKE 10264,60:POKE 10265,126:POKE 10266,126:POKE 10267,126:POKE 10268,126:P
OKE 10269,102:POKE 10270,60
2050 POKE 10271,24:RETURN
2060 POKE 10264,0:POKE 10265,0:POKE 10266,32:POKE 10267,96:POKE 10268,255:POKE 1

```

Normally, when power boating, you would make it a habit to avoid all obstacles, but in this variation you must collide with the yellow buoys. Nevertheless, anything else could cause a wreck, so avoid the shallows (marked with Xs) and the red (for danger!) buoys. Use the joystick to change direction and keep going as long as possible.

# ATARI



```

0289,127:POKE 10270,62
2070 POKE 10271,0:RETURN
3000 Y=Y-1
3010 LOCATE X,Y,Z:IF Z=36 OR Z=34 THEN 5000
3020 IF Z=2 THEN SC=SC+50:GOSUB 4500
3030 COLOR 32:PLOT X1,Y1:COLOR 131:PLOT X,Y:Y1=Y
3040 RETURN
3100 X=X+1:IF X>19 THEN X=0
3110 LOCATE X,Y,Z
3120 IF Z=36 OR Z=34 THEN 5000
3130 IF Z=2 THEN SC=SC+50:GOSUB 4500
3140 COLOR 32:PLOT X1,Y1:COLOR 131:PLOT X,Y:Y1=Y
3150 RETURN
3200 Y=Y+1
3210 LOCATE X,Y,Z
3220 IF Z=34 OR Z=36 THEN 5000
3230 IF Z=2 THEN SC=SC+50:GOSUB 4500
3240 COLOR 32:PLOT X1,Y1:COLOR 131:PLOT X,Y:Y1=Y
3250 RETURN
3300 X=X-1:IF X<0 THEN X=19
3310 LOCATE X,Y,Z
3320 IF Z=34 OR Z=36 THEN 5000
3330 IF Z=2 THEN SC=SC+50:GOSUB 4500
3340 COLOR 32:PLOT X1,Y1:COLOR 131:PLOT X,Y:Y1=Y
3350 RETURN
4000 FOR QW=30 TO 0 STEP -2: SOUND 0,QW,10,12:NEXT QW:SOUND 0,0,0,0:RETURN
5000 DATA 128,196,108,120,120,124,62
5005 POP
5010 DATA 0,128,196,108,120,124,62
5020 DATA 0,0,128,196,108,120,124
5030 DATA 0,0,0,128,196,108,120,120
5040 DATA 0,0,0,128,196,108,120,120
5050 DATA 0,0,0,0,128,196,108,120
5060 DATA 0,0,0,0,0,128,196,108
5070 DATA 0,0,0,0,0,0,128,196
5080 DATA 0,0,0,0,0,0,0,128
5090 RESTORE 5000
5095 COLOR 32:PLOT X1,Y1:COLOR 131:PLOT X,Y
5096 FOR JK=1 TO 20:NEXT JK
5100 FOR JKL=0 TO 8
5110 FOR J=10264 TO 10271:READ JK:POKE J,JK:NEXT J
5120 FOR QW=1 TO 20:NEXT QW:NEXT JKL
5130 FOR JK=1 TO 200:NEXT JK
5140 ? #6: " "
5150 ? #6: "YOU SCORED "SC:" POINTS THAT TIME"
5160 IF SC>HIGH THEN HIGH=SC
5170 ? #6: "HIGH-"HIGH
5180 CH=CH+1:IF CH=5 THEN ? #6: #6: "PLEASE WAIT":GOTO 15
5190 FOR JK=1 TO 300:NEXT JK:GOTO 90

```

# TIMEBOMB BY KIERAN O'DONNELL

In this game, you control a roving pair of pliers, which appear in the centre of the screen. Using the keys:

"T" — Up "F" — Left  
"V" — Down "G" — Right

you must guide the pliers to the bomb which appears. Too easy, I hear you cry! So it's all against the clock which you'll see counting rapidly at the top of the screen. Having defused a bomb, another immediately appears elsewhere on the screen. There are five bombs to defuse on the first screen and an extra bomb appears on each subsequent screen.

Just to make things a little more difficult, the pliers leave behind them a trail of spaces which you won't be able to cross, so you'll have to plan your route quickly and carefully. Fortunately the screen wraps around, so you can go off one side and come on the other, making things a little easier.

The program loads in two sections which you'll have to save, one after the other on tape. When typing in the listing, you'll often find words within square brackets, indicating the keys you should press. This method eliminates all those nasty-looking graphics characters. So:

## PART 1

```
10 POKE52,28:POKE56,28:CLR
20 POKE36879,8:PRINT"[CLEAR]PLEASE WAIT
":PRINT"[DOWN]MAKING GRAPHICS"
30 FORI=7168TO7679:POKEI,PEEK(I+25600):N
EXT
40 I=7432
50 READA:IFA=999THEN150
60 POKEI,A:I=I+1:GOTO50
70 DATA63,67,253,133,181,181,134,252,6,9
,16,56,108,108,108,56
80 DATA80,128,135,244,20,20,247,0,0,0,1
18,85,86,85,117,0
90 DATA60,126,219,231,231,219,126,60,2,5
,62,108,196,12,24,48
100 DATA0,0,112,71,96,71,112,0,31,63,127
,255,128,128,128,157
110 DATA255,253,251,255,7,7,119,138,13
8,138,138,138,128,128,255
120 DATA167,167,167,167,167,6,4,248,8,24
,32,101,161,165,37,37
130 DATA0,0,0,76,210,92,80,76,0,0,225,14
6,229,149,149,226
140 DATA0,0,0,86,117,86,85,86,999
150 PRINT"[CLEAR][WHITE][DOWN][DOWN][DOW
N][DOWN][DOWN]"
160 PRINT"[RVSON][c *][RIGHT][SEP][RIGHT
][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIG
HT][SEP][c *][RIGHT][RIGHT][RIGHT][SEP]
[c *]"
170 PRINT"[RVSON][RIGHT][RIGHT][RIGHT]
[RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIG
HT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][R
IGHT]"
180 PRINT"[RVSON][RIGHT][SEP][c *][
c *][RIGHT][RIGHT][RIGHT][SEP][c *][
RIGHT]"
190 PRINT"[RVSON][SEP][RVSOFF][SEP][RVS
ON][RIGHT][RIGHT][RIGHT][RVSOFF][c *]
[RVSON][c *][RIGHT][RIGHT][RIGHT][c *]
200 PRINT"[RVSON][RVSOFF][c *][RVSON][RV
SON][RVSOFF][c *][RVSON][RVSOFF][RVSON]
[RVSON][RVSOFF][c *][RVSON][RVSOFF][SEP
P][RVSON][RIGHT][RIGHT][RVSOFF][c *][RV
SON]"
210 PRINT"[RVSON][RIGHT][RVSOFF][c *][R
VSON][c *][RIGHT][RIGHT][RIGHT][RIGHT][R
IGHT][RIGHT][RIGHT][RIGHT]"
220 PRINT"[RVSOFF][SEP][RIGHT][RIGHT][RV
SOFF][c *][RVSON]"
```

```
230 PRINT"[DOWN][DOWN]
TS":FORI=1TO5000:NEXT
240 PRINT"[WHITE][CLEAR][RIGHT][RIGHT][c
*][RVSON][RVSOFF][SEP]":PRINT"[RIGHT
][RIGHT][RIGHT][RIGHT][RVSON]"
250 PRINT"[RIGHT][RIGHT][RIGHT][RIGHT][R
VSON][RIGHT][RIGHT][RVSOFF][c *][SEP]
[RVSON][RIGHT]":PRINT"[RVSON][RIGHT][R
IGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIG
HT][RIGHT][RIGHT][c *]":PRINT"[RVSON]
[RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIG
HT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][R
VSOFF][c *][RIGHT][c *][RIGHT][c *][RIGH
T][RIGHT][SEP][RIGHT][c *][RVSON][RVSO
FF][SEP]"
270 PRINT"[RIGHT][RIGHT][DOWN][RVSON][SEP
P][c *][RIGHT][RVSOFF][RIGHT]":PRINT"[
RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIG
HT][RVSON][c *]":PRINT"[RIGHT][RIGHT][RIG
HT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT]
[RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIG
HT][RVSON]"
280 PRINT"[RIGHT][RIGHT][RVSON][RIGHT][
RIGHT][RIGHT][RIGHT][RVSOFF][c *][RIGHT
][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIG
HT][RVSON]":PRINT"[RIGHT][RIGHT][RVSON]
[RVSOFF][c *][I][I][SEP][RIGHT][RVSON]
[SEP][c *][I][I][SEP][c *][RIGHT][RIG
HT]"
290 PRINT"[RIGHT][RIGHT][RVSON][c *][I][
I][c *][RIGHT][RIGHT][RIGHT][RVSOFF][c
*][SEP][RVSON][RIGHT][c *]":FORI=1T
O2:PRINT"[RIGHT][RIGHT][RVSON][RIGHT][R
IGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIG
HT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][R
IGHT]"
300 PRINT"[RIGHT][RIGHT][RVSOFF][c *][RV
SON][RVSOFF][SEP][RIGHT][c *][RVSON][
RVSOFF][SEP][RVSON][RVSOFF][RVSON][RV
SON][RVSOFF][c *][RVSON][RVSOFF][SEP]
310 GETA$:IFA$="THEN310
320 PRINT"[CLEAR][WHITE]":POKE198,1:POKE
631,131
```

## PART 2

```
5 DIMA(30),HS(5),N$(5):FORI=1TO5:HS(I)=1
0:N$(I)="VIC-20":NEXT
10 POKE36878,15:POKE36879,25:S=36875:S=
S+1:N=S-1:SC=0:TB=5:W=1
20 CO=30720:POKE36869,255:PRINT"[CLEAR]"
:YM=7932:C=1:DD=TB:CH=1
```





# SKYWALKER BY M. OFFIELD

You'll certainly need some sort of force with you if you're going to outsmart the Beeb in this skirmish. The idea is that you must deflect the bolts of energy fired from the spinning weapons droid. You do this by swinging your Light Sabre about using: Z — Left, X — Right. Each time you're hit, you lose one of your three lives.

Try the program out on level one, just to get the hang of things. Level five is just what it says — impossible!

Electron owners will be pleased to hear that the game will also work on their machines. Type it in as listed. The teletext title screen will look a little unusual, with double printing and so on. You'll also find the game a little slow, but if you just use the more difficult levels, it'll still be quite a challenge.

```

50 :
60 ONERROR MODE4:REPORT:PRINT" at 11n
@ " ; ERL:END
70 :
80 PROCINIT
90 MODE7
100 VDU23,1,0;0;0;0;0;
110 PROCTITLEPAGE
120 MODE1
130 VDU23,1,0;0;0;0;0;
140 PROCSCREEN
150 PROCPLAY
160 MODE7
170 VDU23,1,0;0;0;0;0;
180 PROCTABLE
190 IF AG$="y" THEN 120
200 PROCBYE
210 MODE7
220 END
230 :
240 DEF PROCINIT
250 *TV255
260 VDU23,224,0,0,7,31,63,63,127,127,2
3,225,0,0,128,224,208,96,168,208,23,226,
127,127,63,63,31,7,0,0,23,227,208,168,96
,208,224,128,0,0
270 BALL$=CHR$224+CHR$225+CHR$10+CHR$8
+CHR$8+CHR$226+CHR$227
280 VDU23,228,0,0,48,56,48,112,120,124
,23,229,114,114,49,49,48,48,56,23,230
,0,6,15,7,6,7,15,63,23,231,63,79,79,159,
159,31,63,63,23,232,0,0,48,57,49,114,122
,126
290 VDU23,233,112,112,48,48,48,48,48,5
6,23,234,0,6,15,135,71,71,111,63,23,235,
63,15,15,31,31,31,63,63,23,236,0,0,0,0,0
,60,126,255
300 DIMFIGHT$(3),HSN$(5),HSX(5)
310 FIGHT$(1)=CHR$232+CHR$234+CHR$10+C
HR$8+CHR$8+CHR$233+CHR$235
320 FIGHT$(2)=CHR$228+CHR$230+CHR$10+C
HR$8+CHR$8+CHR$229+CHR$231
330 FIGHT$(3)=CHR$32+CHR$230+CHR$10+CH
R$8+CHR$8+CHR$236+CHR$231
340 FORIN=1TO5:HSN$(IN)="The Beeb":HSX
(IN)=100:NEXT
350 ENDPROC
360 :
370 DEF PROCTITLEPAGE
380 FORLX=0TO24:PRINTTAB(0,LX)CHR$157C
HR$130:NEXT

```

```

390 FORIN=1TO2:PRINTTAB(8,1-IN)CHR$141
CHR$129CHR$136"S K Y W A L K E R":NEXT
400 PRINTTAB(12,6)CHR$132"BY M.OFFIELD
"
410 PRINTTAB(3,10)CHR$130"MOVE YOUR LI
GHT SABRE USING THE"
420 PRINTTAB(11,12)CHR$130"FOLLOWING K
EYS:-"
430 PRINTTAB(4,14)CHR$130"Z - LEFT"TAB
(25,14)"X - RIGHT"
440 PROCTUNE
450 PRINTTAB(9,17)CHR$133"SELECT LEVEL
(1-5)"
460 PRINTTAB(5,19)CHR$133"1 - EASY ...
5 - IMPOSSIBLE"
470 REPEAT
480 *FX15,1
490 LEV$=GET$:LEVX=VAL(LEV$)
500 UNTIL LEVX<6 AND LEVX>0
510 PRINT TAB(29,17):LEVX
520 PRINTTAB(11,22)CHR$134"Any key to
play"
530 GX=GET
540 ENDPROC
550 :
560 DEF PROCSCREEN
570 P=0:FIREFLAG$=0:SCOREX=0:LIVESX=3
580 VDU 29,640:200;
590 VDU19,1,7;0;19,2,4;0;19,3,0;0;
600 GCOL0,130
610 CLG
620 PROCLUKE
630 *FX15,1
640 GCOL0,1:MOVE-640,-200:DRAW640,-200
:PLOT85,-640,-100:PLOT85,640,-100
:650 MOVE 340,-100:PLOT85,640,50:MOVE -
340,-100:MOVE -640,-100:PLOT85,-640,50
660 VDU4 -
670 COLOUR1
680 PRINTTAB(10,0)"SCORE"TAB(23,0)"LIV
ES"

```

BB1

```
690 VDU5
700 GCOL0,0:MOVE(-18.5*32),-140:PRINT"
MAY THE
710 GCOL0,2:MOVE(-5.5*32),-140:PRINT"B
E E B"
720 GCOL8,0:XBK=0:YBK=600:MOVINGK=0:MO
VEXBK,YBK:PRINTBALLS
730 GCOL8,0
740 SOUND0,-15.7.50
750 FOR FX=50 TO 250 STEP 4
760 SOUND1,0,FX,1
770 PLOT69,0,FX,100
780 NEXT
790 ENDPROC
800 :
810 DEF PROCPLAY
820 REPEAT
830 PROCSCORE
840 *FX15,1
850 IF INKEY(-98) FX=1:PROCSABRE
860 IF INKEY(-67) FX=0:PROCSABRE
870 PROCMOVEBALL
880 IF RND(12-LEVX)=1 AND FIREFLAGN=0
PROCBALLFIRE
890 UNTIL LIVESX=0
900 ENDPROC
910 :
920 DEF PROCLUKE
930 GCOL0,0:MOVE0,200:DRAW30,190:PLOT8
5,-30,190
940 RESTORE 1010
950 FOR COORDS=1 TO 14
960 READ XN,YN
970 PLOT 85,XN,YN:PLOT 85,-XN,YN
980 NEXT
```

# SKYWALKER BY M. OFFIELD

```

990 ENDPROC
1000 :
1010 DATA 40,170,35,150,30,130,20,110,4
0,100,30,100,50,90,70,80,90,40,95,0,100,
-20,90,-30,60,-30,55,-100
1020 :
1030 DEF PROCsabre
1040 GCOL4,0:MOVE0,0:DRAW (P*250),(COS(
P)*350)
1050 IF FX=1 P=P-0.1:IF P<-PI/2 P=P+0.1
1060 IF FX=0 P=P+0.1:IF P>PI/2 P=P-0.1
1070 PROCmoveball
1080 GCOL4,0:MOVE 0,0:DRAW (P*250),(COS
(P)*350)
1090 ENDPROC
1100 :
1110 DEF PROCmoveball
1120 IF MOVINGX=FALSE AND RND(5)<4 THEN
1230
1130 GCOL0,2:MOVE XBN,YBN:PRINTBALL$
1140 IF MOVINGX=TRUE THEN 1180
1150 MOVINGX=TRUE
1160 DRN=RND(4):DSX=200+RND(150)
1170 XBN=XBN+DSX:YBN=YBN+DSX
1180 IF (DRN=1 AND XBN>550) OR (DRN=2 A
ND XBN<-580) OR (DRN=3 AND YBN>730) OR (
DRN=4 AND YBN<450) THEN MOVINGX=FALSE:EN
DPROC
1190 IF DRN=1 THEN XBN=XBN+20:IF XBN>XB
1X THEN MOVINGX=FALSE:ENDPROC
1200 IF DRN=2 THEN XBN=XBN-20:IF XBN<XB
2X THEN MOVINGX=FALSE:ENDPROC
1210 IF DRN=3 THEN YBN=YBN+20:IF YBN>YB
1X THEN MOVINGX=FALSE:ENDPROC
1220 IF DRN=4 THEN YBN=YBN-20:IF YBN<YB
2X THEN MOVINGX=FALSE:ENDPROC
1230 GCOL0,1:MOVE XBN,YBN:PRINTBALL$
1240 ENDPROC
1250 :
1260 DEF PROCballfire
1270 SOUND0,-15,7,4
1280 FOR FX=255 TO 95 STEP -40:SOUND1,-
5,FX,1:NEXT
1290 X1X=XBN+30:Y1X=YBN-30
1300 FIREFLAGN=1
1310 IF X1X=0 M=0:GOTO1330
1320 M=Y1X/X1X
1330 GCOL0,1
1340 FOR YX=Y1X TO Y1X-30 STEP -(LEVX*5
)
1350 IF M=0 XX=X0:GOTO1370
1360 XX=YX/M
1370 PLOT 69,XX,YX
1380 NEXT
1390 REPEAT
1400 IFRND(5)<4 GOTO 1430
1410 *FX15,1
1420 IF INKEY(-98) FX=1:PROCSABRE ELSE
IF INKEY(-67) FX=0:PROCSABRE ELSE PROCMA
VEBALL
1430 IF M=0 XX=X0:GOTO1450
1440 XX=YX/M
1450 HIT1X=POINT(XX-5,YX):HIT2X=POINT(X
N,YX):HIT3X=POINT(XX+5,YX)
1460 GCOL0,1

```

```

1470 PLOT 69,XN,YN
1480 Y1X=YX+(LEVX*25)
1490 IF M=0 X1X=0:GOTO1510
1500 X1X=Y1X/M
1510 GCOL0,2
1520 PLOT 69,X1X,Y1X
1530 HITSABREX=(HIT1X=1) OR (HIT2X=1) O
R (HIT3X=1)
1540 DEADN=(HIT1X=0) OR (HIT2X=0) OR (H
IT3X=0)
1550 YX=YX-(LEVX*5)
1560 UNTIL HITSABREX OR DEADN
1570 YX=YX-(LEVX*5)
1580 FIREFLAGN=0
1590 GCOL0,1:MOVE XBN,YBN:PRINTBALL$
1600 PROCCOVERSHOT
1610 IF HITSABREX THEN PROCHITSABRE ELS
E IF DEADN THEN PROCDEAD
1620 ENDPROC
1630 :
1640 DEF PROCCOVERSHOT
1650 GCOL4,0:MOVE0,0:DRAW (P*250),(COS(
P)*350)
1660 GCOL0,2
1670 FOR YLX=YX TO YX+(LEVX*20) STEP (L
EVX*5)
1680 IF M=0 THEN XLX=0:GOTO 1700
1690 XLX=YLX/M
1700 PLOT69,XLX,YLX
1710 NEXT
1720 GCOL4,0:MOVE0,0:DRAW (P*250),(COS(
P)*350)
1730 ENDPROC
1740 :
1750 DEF PROCHITSABRE
1760 SCOREX=SCOREN+10
1770 SOUND0,-15,7,4
1780 FOR FX=95 TO 215 STEP 30:SOUND1,-2
,FX,1:NEXT
1790 ENDPROC
1800 :
1810 DEF PROCDEAD
1820 LIVESN=LIVEN-1
1830 SOUND 0,-15,7,125
1840 FOR FX=250 TO 0 STEP -2:SOUND1,0,F
X,1:NEXT
1850 TIME=0:REPEATUNTILTIME>50
1860 GCOL0,1:MOVE-640,-200:DRAW640,-200
:PLOT85,-640,-100:PLOT85,640,-100
1870 XFX=14.5*32:YFX=-100
1880 FOR BN=1 TO 5
1890 FOR CHX=2 TO 1 STEP -1
1900 GCOL0,0
1910 MOVE (-XFX-64),YFX:PRINT FIGHT$(CH
X)
1920 MOVE XFX,YFX:PRINT FIGHT$(CHX)
1930 SOUND 0,-15,3,CHX,5
1940 TIME=0:REPEAT UNTIL TIME>20
1950 GCOL0,1
1960 MOVE (-XFX-64),YFX:PRINT FIGHT$(CH
X)
1970 MOVE XFX,YFX:PRINT FIGHT$(CHX)
1980 NEXT:NEXT
1990 GCOL0,0
2000 MOVE (-XFX-64),YFX:PRINT FIGHT$(3)
2010 MOVE XFX,YFX:PRINT FIGHT$(3)

```

# 880

```

2020 SOUND 0,-15,6,30
2030 TIME=0:REPEAT UNTIL TIME>300
2040 PROCDITTY
2050 IF LIVESX=0 THEN 2110
2060 GCOL0,1
2070 MOVE (-FXN-64).YFN:PRINT FIGHT$(3)
2080 MOVE XFN,YFN:PRINT FIGHT$(3)
2090 GCOL0,0:MOVE (-14.5*32),-140:PRINT
"MAY THE BE WITH YOU"
2100 GCOL0,2:MOVE (-5.5*32),-140:PRINT"
B E B"
2110 ENDPROC
2120 :
2130 DEF PROCSCORE
2140 VDU4
2150 COLOUR1
2160 PRINTTAB(16,0):SCORENTAB(29,0):LIV
ESX
2170 VDU5
2180 ENDPROC
2190 :
2200 DEF PROCTUNE
2210 RESTORE 2320
2220 FOR SDX=1TO12:READ FX,DX:SOUND1,-1
5,FX,DX:NEXT
2230 RESTORE 2330
2240 FOR SDX=1TO4:READ FX,DX:SOUND1,-15
,FX,DX:NEXT
2250 RESTORE 2320
2260 FOR SDX=1TO12:READ FX,DX:SOUND1,-1
5,FX,DX:NEXT
2270 RESTORE 2340
2280 FOR SDX=1TO4:READ FX,DX:SOUND1,-15
,FX,DX:NEXT
2290 TIME=0:REPEATUNTILTIME>400
2300 ENDPROC
2310 :
2320 DATA1.10,109.20,101.3,97.3,89.3,1
29.10,109.20,101.3,97.3,89.3,129.10,109.
20
2330 DATA101.3,97.3,101.3,89.40
2340 DATA121.3,117.3,121.3,109.40
2350 :
2360 DEF PROCDITTY
2370 RESTORE 2420
2380 FORSDX=1TO7:READ FX,DX:SOUND1,-15,
FX,DX:NEXT
2390 TIME=0:REPEATUNTILTIME>500
2400 ENDPROC
2410 :
2420 DATA81.10,105.21,101.3,93.3,89.3,1
25.10,109.30
2430 :
2440 DEF PROCTABLE
2450 PROCNEWHIGH:CLS
2460 FORIX=1TO2:PRINTTAB(5,1+IX)CHR$129
CHR$136CHR$141"H I G H S C O R E S":NE
XT
2470 PRINTTAB(14,5)CHR$132"LEVEL ":LEVX
2480 FORNX=1TO5
2490 PRINTTAB(9,7+NX*2)CHR$(129-NX)HSN$(
NX)TAB(24,7+NX*2):HSN$(NX)
2500 NEXT
2510 *FX15,1
2520 PRINTTAB(10,21)CHR$129"ANOTHER GAM

```

```

2530 REPEAT:AG$=GET$:UNTIL AG$="Y" OR A
GS="N"
2540 PRINT TAB(25,21)AG$
2550 TIME=0:REPEATUNTILTIME>100
2560 ENDPROC
2570 :
2580 DEF PROCNEWHIGH
2590 IF SCOREX<=HSN(5) THEN ENDPROC
2600 FORIN=1TO2
2610 PRINTTAB(1,2+IN)CHR$129CHR$137CHR$
136CHR$141"C O N G R A T U L A T I O N S
"
2620 NEXT
2630 PRINTTAB(9,10)CHR$130"YOUR SCORE I
S IN THE"
2640 PRINTTAB(14,12)CHR$131"TOP FIVE !!
"
2650 PROCCONGRATS
2660 PRINTTAB(8,16)CHR$132"PLEASE INPUT
YOUR NAME"
2670 PRINTTAB(0,20)CHR$133
2680 *FX15,1
2690 INPUT TAB(15,20)NAME$
2700 PRINTTAB(25,20)CHR$152
2710 FOR LX=21 TO 23:PRINTTAB(0,LX)CHR$
152:NEXT
2720 TIME=0:REPEAT UNTIL TIME>300
2730 IF LEN(NAME$)>10 THEN NAME$=LEFT$(
NAME$,10)
2740 REM BUBBLE SORT ROUTINE
2750 HSN(5)=SCOREX:HSN$(5)=NAME$
2760 FOR LOOPN=4 TO 1 STEP -1
2770 CX=0
2780 FOR IX=1 TO LOOPN
2790 IF HSN(IX)>HSN(IX+1) THEN CX=CX+1
:GOTO 2830
2800 DUMX=HSN$(IX+1):DUMX=HSN(IX+1)
2810 HSN$(IX+1)=HSN$(IX):HSN(IX+1)=DUMX(
IX)
2820 HSN$(IX)=DUMX:HSN(IX)=DUMX
2830 NEXTIX
2840 IF CX=LOOPN THEN LOOPN=0
2850 NEXTLOOPN
2860 ENDPROC
2870 :
2880 DEF PROCCONGRATS
2890 RESTORE 2940
2900 FORSDX=1TO22:READ FX,DX:SOUND1,-15
,FX,DX:NEXT
2910 TIME=0:REPEATUNTILTIME>300
2920 ENDPROC
2930 :
2940 DATA81.5,89.5,97.5,101.10,81.15,10
1.5,97.5,101.5,109.10,89.15,89.5,97.5,10
1.5,117.9,109.3,109.3,101.5,101.5,97.5,8
9.5,97.5,101.25
2950 :
2960 DEF PROCBYE
2970 CLS
2980 FOR YGBN=3 TO 18 STEP 15
2990 FOR IX=1TO2:PRINTTAB(10,YGBN-IX)CH
R$141CHR$129CHR$136"G O O D B Y E":NEXT
3000 NEXT
3010 PRINTTAB(6,9)CHR$132"THE HIGHEST S
CORE WAS"CHR$136:HSN(1)
3020 PRINTTAB(8,12)CHR$130"ACHIEVED BY"
CHR$136HSN$(1)
3030 PRINTTAB(14,14)CHR$131"ON LEVEL"CH
R$136:LEVX
3040 RESTORE 3090
3050 FORSDX=1TO16:READ FX,DX:SOUND1,-15
,FX,DX:NEXT
3060 GX=INKEY(1000)
3070 ENDPROC
3080 :
3090 DATA89.10,89.6,125.20,89.10,97.6,1
29.20,125.7,117.3,137.10,129.7,117.3,137
,10,129.7,117.3,125.6,109.30

```

# CATERPILLAR BY STEVEN HOWARD

```

10 TX=0:SY=0
20 SYMBOL AFTER 240
30 SYMBOL 240,153,90,24,219,90,2
19,90,219
40 SYMBOL 241,6,24,126,223,191,1
91,223,126
50 SYMBOL 242,60,126,255,255,24,
24,24,24
60 SYMBOL 243,0,0,0,15,63,127,25
5,255
70 SYMBOL 244,0,0,0,0,224,240,24
0,248
80 SYMBOL 245,7,7,7,7,0,0,0
90 SYMBOL 246,8,28,28,107,127,10
7,8,28
100 SYMBOL 247,128,112,248,252,2
54,126,31,7
110 SYMBOL 248,133,201,113,49,11
9,30,4,4
120 SYMBOL 249,0,24,44,94,94,191
,191,255
130 SYMBOL 250,0,255,126,60,7,0,
0,0
140 GOSUB 780
150 INK 0,0:INK 1,10:INK 2,12:IN
K 3,4:INK 4,3:INK 5,9:INK 6,13:1
NK 7,9,0
160 MODE 0
170 TAG
180 X%=290:Y%=100
190 TI=0
200 SOUND 1,300,10,7,0,0,1
210 GOSUB 380
220 GOSUB 420
230 TI=TI+20
240 B%=247:C%=4
250 IF TI>1000 AND TI<1500 THEN
GOSUB 520
260 B%=248:C%=6
270 IF TI>3000 AND TI<3500 THEN
GOSUB 520
280 B%=242:C%=5
290 IF TI>3500 AND TI<4500 THEN
GOSUB 520
300 B%=246:C%=3
310 IF TI>5000 AND TI<5500 THEN
GOSUB 520
320 B%=241:C%=1
330 IF TI>6500 AND TI<7000 THEN
GOSUB 520
340 IF INT(RND*50)=1 THEN GOSUB
570
350 GOSUB 640
360 IF TI>7500 THEN TI=0:GOTO 20
0
370 GOTO 200

```

Well, it would seem that caterpillars don't like mushrooms. So, in this game you must guide your caterpillar through the mushroom patch, avoiding the mushrooms, but picking up twigs, flowers, apples and acorns for bonus points as you work your way through the seasons.

There are only two keys to master: "Z"—Left "M"—Right but you'll still find this game pretty tough. For such a short game, the graphics are excellent and will no doubt provide many tips for prospective game writers.

```

380 MOVE X%,Y%:PRINT CHR$(240);
390 IF INKEY(38) AND X%>20 THEN
X%=X%-8
400 IF INKEY(71) AND X%<600 THEN
X%=X%+8
410 RETURN
420 TAGOFF
430 L%=INT(RND*17)+1
440 PEN 5
450 LOCATE L%,1:PRINT CHR$(243)
460 LOCATE L%+1,1:PRINT CHR$(244)
)
470 PEN 2
480 LOCATE L%,2:PRINT CHR$(245)
490 LOCATE 1,1:PRINT CHR$(11)
500 TAG
510 RETURN
520 TAGOFF
530 PEN C%
540 LOCATE (INT(RND*20))+1,1:PRI
NT CHR$(B%)
550 TAG
560 RETURN
570 TAGOFF

```



# AMSTRAD CPC 464

```

580 O%=INT(RND*18)
590 PEN 7
600 LOCATE O%+1,1:PRINT CHR$(249
)
610 LOCATE O%+1,2:PRINT CHR$(250
)
620 TAG
630 RETURN
640 H%=%X+16
650 V%=%Y
660 IF TEST(H%,V%)=4 THEN S%=%+
5:SOUND 1,80,10,7
670 IF TEST(H%,V%)=6 THEN S%=%+
10:SOUND 1,70,10,7
680 IF TEST(H%,V%)=3 THEN S%=%+
15:SOUND 1,60,10,7
690 IF TEST(H%,V%)=1 THEN S%=%+
20:SOUND 1,50,10,7
700 IF TEST(H%,V%)=7 THEN S%=%+
50:SOUND 1,40,10,7
710 IF TEST(H%,V%)=5 THEN GOTO 7
50
720 TAGOFF
730 TAG
740 RETURN
750 SOUND 1,650,50,7,0,0,1
760 IF S%>T% THEN T%=S%
770 FOR O=1 TO 2000:NEXT
780 INK 0,0:INK 1,12:INK 2,4:INK
3,9
790 MODE 1:PEN 2:BORDER 0:PAPER
0
800 LOCATE 14,1:PRINT "CATERPIL
LAR"
810 PEN 3:LOCATE 4,4:PRINT "Guid
e the caterPillar through the"
820 PRINT " mushroom Patch.You

```

```

start in Autumn"
830 PRINT " and work your way t
hrough the four"
840 PRINT " seasons.You score p
oints by eating"
850 PRINT " food on the way:"
860 PEN 1:LOCATE 10,11:PRINT "le
aves.....5 Points"
870 PRINT "
Points" twigs.....10
880 PRINT "
Points" flowers...15
890 PRINT "
Points" apples....20
900 PRINT "
Points" acorns....50
910 PEN 2:LOCATE 10,17:PRINT "Z
= left M = right"
920 PEN 3:LOCATE 11,19:PRINT "HI
GH SCORE ";T%
930 LOCATE 11,21:PRINT "YOUR SCO
RE ";S%
940 PEN 1:LOCATE 10,24:PRINT "PR
ESS SPACE TO START";
950 S%=0
960 G=INKEY(47):IF G=0 THEN GOTO
150 ELSE GOTO 960

```

# SHOWJUMPING BY ERIC TILLEY

```

1 GRAPHICS 2:16:POSITION 4,0:7 #6:"SHOW JUMPING":POSITION 8,1:7 #6:"BY":POSITION
4,2:7 #6:"ERIC TILLEY"
2 GOSUB 2000:POSITION 4,4:7 #6:"[INV]MOVIE[INV] [INV]STICK[INV] [INV]TO[INV]":POS
ITION 4,5:7 #6:"[INV]SELECT[INV] [INV]NUMBER[INV]":POSITION 6,5:7 #6:"[INV]OF[INV]
V [INV]JUMPS[INV]"
3 POSITION 2,8:7 #6:"trigger to start":NUMJMP=10
4 POSITION 2,11:PRINT #6:"JUMPS=":NUMJMP:FOR D=1 TO 50:NEXT D
5 IF STICK(0)=0 THEN 10
6 IF STICK(0)<15 THEN NUMJMP=NUMJMP+10:IF NUMJMP>30 THEN NUMJMP=10
7 GOTO 4
10 GRAPHICS 18:POSITION 2,2:7 #6:"PLEASE WAIT":POSITION 2,4:7 #6:"WHILE STEWARDS
"POSITION 2,6:7 #6:"SET UP COURSE"
11 FOR D=1 TO 1000:NEXT D
12 GOSUB 10000
13 UP=0:UPC=1:SP=50:H=1:FLTS=0
14 JMPCCU=0
20 D=USR(VP,REST1,0,PL0):D=USR(VP,REST2,0,PL1)
22 FOR D=1 TO 200:NEXT D
24 GOSUB 1540
26 STATIM=(PEEK(18)*65536)+(PEEK(19)*256)+(PEEK(20))/60
40 J=INT(RND(0)*10)+1:D=USR(VP,JUMP(J),0,PL2):IF J=1 OR J=4 THEN POKE 53250,3
50 FOR I=191 TO 39 STEP -4
51 IF UP=0 THEN SOUND 0,200,8,15: SOUND 0,0,0,0
55 IF STRIG(0)=0 AND UP<1 THEN UP=1:D=USR(VP,HORSE(3),PL0,PL0-10):D=USR(VP,HORSE
K(2),PL1,PL1-10):DST=10-(SP/10)
56 IF UP=0 THEN D=USR(VP,HORSE(H),PL0,PL0):D=USR(VP,HORSE(H+1),PL1,PL1)
57 IF UP=1 THEN UP=UPC+1:IF UP<DST THEN UP=10:UP=0:D=USR(VP,HORSE(H),PL0-10,PL
0):D=USR(VP,HORSE(H+1),PL1-10,PL1)
58 IF UP=0 THEN SOUND 0,130,8,15: SOUND 0,0,0,0
60 POKE 53250,I
85 IF PEEK(HIT1)>2 OR PEEK(HIT2)>2 THEN GOSUB 300
90 H=H+2:IF H>3 THEN H=1
92 IF STICK(0)>8 THEN SP=SP-2:IF SP<0 THEN SP=0
93 IF STICK(0)>8 AND STICK(0)<13 THEN SP=SP+2:IF SP>50 THEN SP=50
95 FOR D=1 TO SP:NEXT D
100 NEXT I
101 JMPCCU=JMPCCU+1:IF JMPCCU=NUMJMP THEN 600
105 POKE 53250,INT(RND(0)*2)
110 D=USR(VP,JUMP(J),PL2,0)
120 GOTO 40
300 SOUND 0,2,8,10:FLTS=FLTS+4:SP=50:UP=0:UPC=1
306 D=USR(VP,HORSE(3),PL0-10,0):D=USR(VP,HORSE(2),PL1-10,PL1)
310 D=USR(VP,FALL1,PL0,PL0):D=USR(VP,FALL2,PL1,PL1)
330 FOR D=1 TO 1-40 STEP -1
340 POKE 53250,I
350 NEXT I
360 I=I-40:POKE HITCLR,I
370 SOUND 0,0,0,0:FOR D=1 TO 250:NEXT D
380 RETURN
600 FINTIM=(PEEK(18)*65536)+(PEEK(19)*256)+(PEEK(20))/60
610 D=USR(VP,REST1,PL0,PL0):D=USR(VP,REST2,PL1,PL1)
620 GOSUB 1540
630 TOTIM=FINTIM-STATIM
640 GRAPHICS 2:16
641 POKE 53277,0
660 POSITION 0,0:PRINT #6:"[INV]JUMPS[INV] ":NUMJMP
660 POSITION 0,2:PRINT #6:"[INV]faults[INV] ":FLTS
670 POSITION 0,4:PRINT #6:"time ":TOTIM
676 GOSUB 2000
700 GOTO 700
1540 FOR J=1 TO 25
1550 SOUND 0,10,14,15: SOUND 1,20,14,15
1560 FOR D=1 TO 5:NEXT D
1570 SOUND 0,15,14,15: SOUND 1,25,14,15
1580 FOR D=1 TO 5:NEXT D
1590 NEXT J
1600 SOUND 0,0,0,0: SOUND 1,0,0,0
1610 RETURN
2000 FOR S1=0 TO 2
2010 FOR S2=1 TO 2
2020 GOSUB 2070:FOR D=1 TO 60:NEXT D:GOSUB 2120
2030 FOR S3=1 TO 3:GOSUB 2070:FOR D=1 TO 15:NEXT D:GOSUB 2120:NEXT S3
2040 NEXT S2
2045 NEXT S1
2050 S1=3:GOSUB 2070:FOR D=1 TO 200:NEXT D:GOSUB 2120
2060 RETURN
2070 SOUND 0,81,10,10
2080 IF S1>0 THEN SOUND 1,64,10,10
2090 IF S1>1 THEN SOUND 2,53,10,10
2100 IF S1>2 THEN SOUND 3,40,10,10

```

Make the jumps safely, but all the time keep an eye on the clock. This game has you competing as a show jumper.

You have to clear all the jumps, including uprights, doubles and the dreaded water, to complete the course.

The game graphics are excellent, especially when your horse falls! Of course, the idea is that

```

2110 RETURN
2120 SOUND 0,0,0,0: SOUND 1,0,0,0: SOUND 0,0,0,0
2130 RETURN
10000 REM SET UP P.M
10010 POKE 106,PEEK(106)-5:GRAPHICS 3:
10020 FBASER=[PEEK(106)+1]*256
10030 POKE 54279,PEEK(106)+1
10035 POKE 53277,3:POKE 559,46

```





you fall as little as possible, thus keeping down the number of faults at the end. This involves timing the speed and the jumps correctly. Use the joystick:  
RIGHT—Increase Speed  
LEFT—Decrease Speed  
FIRE—Jump

When typing in the listing, take care of the graphics commands, shown as [INV].

ND 2,0,0,0;SOUND 3,0,0,0

3+16

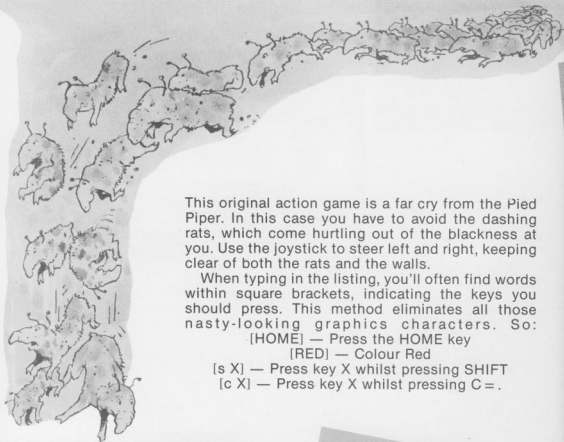


```

10036 FOR I=0 TO 1023
10037 POKE PMBASE+I,0
10038 NEXT I
10040 PLO=PMBASE+580;PL1=PL0+128;PL2=PL1+132
10060 DIM HORSE(4),JUMP(10),TXT$(8)
10070 HORSE(1)=PMBASE;HORSE(2)=HORSE(1)+13;HORSE(3)=HORSE(2)+13;HORSE(4)=HORSE(3)+13
10090 JUMP(1)=HORSE(4)+13;FOR I=0 TO 10:JUMP(I)=JUMP(I-1)+9:NEXT I
10091 FALL1=JUMP(10)+11;FALL2=FALL1+13
10092 REST1=FALL2+13;REST2=REST1+13
10094 RESTORE 10220
10100 FOR I=0 TO 195
10110 READ X:POKE PMBASE+I,X:SOUND 0,1,0,5
10120 NEXT I
10130 POKE 623,1:POKE 712,154:POKE 708,194
10140 COLOR 1
10145 SOUND 0,0,0,0
10150 FOR I=16 TO 23
10160 PLOF 0,I:DRAWTO 39,I
10170 NEXT I
10180 POKE 53248,80:POKE 53249,88
10190 POKE 704,34:POKE 705,34:POKE 706,0
10192 HITCLR=53279:HIT1=53260:HIT2=53261
10220 DATA 12,0,0,0,32,33,35,63,127,127,127,113,41,28
10230 DATA 12,48,116,199,247,204,252,252,248,248,56,208,96
10240 DATA 12,0,0,0,129,67,63,63,127,127,89,209,160
10250 DATA 12,48,112,182,242,207,255,255,248,248,12,138,5
10260 DATA 8,0,0,0,129,129,129,129,129,129
10270 DATA 8,0,0,0,24,25,25,31,31
10280 DATA 8,4,14,26,18,80,126,66,194
10290 DATA 8,129,129,129,129,129,129,129,129,129
10300 DATA 8,12,12,71,36,31,31,63,255
10310 DATA 8,32,64,129,143,255,243,252,255
10320 DATA 8,255,255,129,255,129,255,255,255
10330 DATA 8,24,24,255,129,255,129,255,129
10340 DATA 8,0,0,1,3,7,15,31,63
10350 DATA 10,0,0,0,0,0,0,255,255,255
10360 DATA 12,0,0,0,0,12,12,71,36,31,63,255
10370 DATA 12,0,0,0,0,32,64,129,143,255,243,252,255
10380 DATA 12,1,1,1,3,8,199,63,127,127,127,65,33,32
10400 DATA 12,128,136,14,236,152,249,248,248,240,8,144,16
20000 VP=1636
20005 RESTORE 20050
20010 FOR I=0 TO 43
20020 READ X:POKE VP+I,X
20030 NEXT I
20040 RETURN
20050 DATA 104,162,5,104,149,220,202,16,290,198,220,198,222,160,0,177,224,170
20060 DATA 168,165,223,240,9,169,0,145,222,136,208,249,138,168,168,221,240,7,177,224,145,220,136,208,249,96

```

# SEWER RATS BY MARK ELLIS



This original action game is a far cry from the Pied Piper. In this case you have to avoid the dashing rats, which come hurtling out of the blackness at you. Use the joystick to steer left and right, keeping clear of both the rats and the walls.

When typing in the listing, you'll often find words within square brackets, indicating the keys you should press. This method eliminates all those nasty-looking graphics characters. So:

[HOME] — Press the HOME key

[RED] — Colour Red

[s X] — Press key X whilst pressing SHIFT

[c X] — Press key X whilst pressing C = .

```

1010 PRINTCHR$(147)
1020 L$="":FORI=1TO4:L$=L$+CHR$(157):NEX
T
1030 POKE53280,0:POKE53281,0
1040 GOSUB1820:REM TITLE
1050 REM *** READ SPRITE DATA ***
1060 FORI=0TO62:READJ:POKE832+I,J:NEXT
1070 FORI=0TO62:READJ:POKE896+I,J:NEXT
1080 FORI=0TO62:READJ:POKE960+I,J:NEXT
1090 PRINTCHR$(147)
1100 REM *** SET UP SPRITES ***
1110 POKE2040,13:VC=53248:POKEVC+21,5
1120 POKE2041,14:POKE2042,15:POKEVC+40,8
1130 POKEVC+39,14:POKEVC+23,7:POKEVC+29,
7
1140 POKEVC+0,150:POKEVC+1,180
1150 POKEVC+28,2:POKEVC+41,10
1155 POKE58296,15
    
```

```

1160 REM *** SET UP SOUND ***
1170 S=54272:FORI=0TO24:POKE+S,I,0:NEXT
1180 S1=54286:POKE+S,5,31:POKE+S,6,240
1190 POKE+S,24,15:POKE+S1+5,31:POKE+S1+6,24
0
1200 L=12:X=150:W=12:Z=0:SE=0
1210 POKE646,14
1220 PRINTCHR$(19)CHR$(5)
1230 FORI=1TO21
1240 PRINTTAB(9)"[c B]"SPC(16)"[c B]"
1250 NEXT:POKEVC+31,0:POKEVC+30,0
1280 PRINTCHR$(19)CHR$(30)
1290 FORI=1TO21
1300 PRINTCHR$(18)"SPC(20):
1310 PRINT"
1320 NEXTI
1330 PRINTCHR$(19)CHR$(5)"SCORE:"
1340 PRINTCHR$(19)TAB(28)"HIGH:"
    
```

# CBM 64

```

1350 REM ***MAIN LOOP ***
1360 IFCR=0THENGOSUB1580
1370 SE=SE+1
1380 CY=CY+16:IFCY>210THENCY=1:CR=0
1390 P=PEEK(56320)
1400 IFF=123THENZ=1
1410 IFF=119THENZ=2
1420 ONZGOSUB1530,1550
1430 POKEVC+0,X
1440 POKEVC+4,R:POKEVC+5,CY
1450 PRINTCHR$(19)TAB(6)" "L$SE
1460 IFSE>HITHENHI=SE
1470 PRINTCHR$(19)TAB(34):HI
1480 IFPEEK(VC+31)AND1=1THEN1600
1490 IFPEEK(VC+30)AND1=1THEN1600
1500 POKEVC+31,0:POKEVC+30,0
1510 GOTO1360
1520 REM *** MAN DICTION ***
1530 X=X-2:IFX<10THENX=10
1540 RETURN
1550 X=X-2:IFX>250THENX=249
1560 RETURN
1570 REM *** NEW POSITION OF RAT ***
1580 R=INT(RND(TI)*84+110):CR=2
1590 FORK=150TO250STEP15:POKES+11,17:POK
ES+8,K: NEXT
1583 POKES+11,0:RETURN
1590 REM *** CRASH INTO KERB ***
1600 POKEVC+21,6:POKES1+4,0
1610 POKEVC+2,X:POKEVC+3,180
1620 FORV=15TO0STEP-1:POKES+1,2+V
1630 POKE53270,INT(RND(1)*8)
1640 POKES+4,129:POKES+24,V
1650 POKEVC+37,INT(RND(TI)*15)
1660 POKEVC+38,INT(RND(TI)*15)
1670 NEXT:POKES+4,0
1680 POKEVC+21,0
1690 REM *** NEW GAME ***
1700 PRINTCHR$(19):CY=1:CR=0
1710 POKE646,INT(RND(TI)*15)
1720 FORI=1TO4:PRINTCHR$(17):NEXT
1730 POKE53270,200
1740 PRINTTAB(12)"YOU HAVE BEEN"
1750 PRINTTAB(11)"EATEN PRESS FIRE"
1760 POKEVC+4,R:POKEVC+5,CY
1770 IFPEEK(56320)<>111THEN1700
1780 PRINTCHR$(145)CHR$(145)CHR$(145)

```



```

1790 PRINTTAB(12)"
1800 PRINTTAB(11)"
13
1810 GOTO1110
1820 REM *** TITLE PAGE ***
1830 PRINTCHR$(19)CHR$(5)
1840 PRINTTAB(7)"> S E W E R * ";
1850 PRINT"R A T S <"
1860 PRINTCHR$(17)CHR$(17)
1865 PRINTTAB(5)"
1866 PRINTTAB(5)" BY M.ELLIS"
|[c T][c T][c T][c T][c T][c T][c T]
1870 PRINTTAB(5)"USE JOYSTICK TO ";
1880 PRINT"STEER YOUR MAN"
1890 PRINTTAB(5)"AND AVOID THE ";
1900 PRINT"SEWER RATS...."
1910 PRINTTAB(5)"ALSO WATCH OUT, ";
1920 PRINT"FOR THE WALLS...."
1930 PRINTCHR$(17)CHR$(17)CHR$(17)
1940 PRINTTAB(9)"PRESS A KEY TO PLAY"
1950 GETA$:IFAS=""THEN1950
1960 RETURN:REM INTO GAME
1970 REM *** SPRITE 0 (MAN) ***
1980 DATA0,0,0,0,60,0,0,60,0,0
1990 DATA60,0,0,24,0,0,255,0,1
2000 DATA189,128,1,60,128,1,60,128,1,255
2010 DATA128,0,126,0,0,60,0,0,36
2020 DATA0,0,36,0,0,36,0,0,231,0
2030 DATA0,0,0,0,0,36,0,0,0,0
2040 DATA0,0,0,0,0,0,0,0,0,0
2050 REM ** SPRITE 1 (CROSSBONES) **
2060 DATA0,248,0,1,252,0,3,254,0,2
2070 DATA114,0,2,114,0,3,222,0
2080 DATA3,222,0,3,254,0,1,140,0,2
2090 DATA250,0,6,115,0,2,4,0,0
2100 DATA136,0,0,115,0,0,32,0
2110 DATA0,80,0,0,136,0,1,4,0
2120 DATA6,3,0,2,2,0,0,0,0
2130 REM *** SPRITE 2 (RAT) ***
2140 DATA0,16,0,0,16,0,0,16,0,0
2150 DATA16,0,0,16,0,0,16,0,0
2160 DATA16,0,0,16,0,0,124,0,0,254
2170 DATA0,254,0,0,254,0,0,254
2180 DATA0,254,0,0,254,0,0,186,0
2190 DATA0,254,0,0,124,0,0,56,0,0,16,0,0
,0,0

```

":REM 11  
":REM



# ROCK FALL

BY ANDREW CLAPHAM

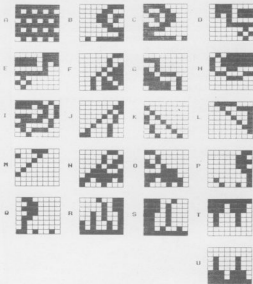
```

1 LET h$="000"
5 REM ROCK FALL February 85
  By A.CLAPHAM
10 DATA 255,170,255,85,255,170
,255,85
20 DATA 7,12,27,26,12,7,1,95
30 DATA 240,152,108,44,152,240
,192,253
40 DATA 112,119,87,16,31,2,5,2
50 DATA 7,247,245,4,252,32,80,
32
60 DATA 3,4,5,11,23,43,45,46
70 DATA 0,224,32,160,220,226,2
50,122
80 DATA 95,191,191,152,99,28,0
,0
90 DATA 250,237,5,237,122,194,
92,32
100 DATA 1,3,5,10,18,36,68,200
110 DATA 0,128,64,160,144,72,68
,34
120 DATA 120,39,17,9,5,3,1,0
130 DATA 28,136,16,32,64,128,0,
0
140 DATA 6,9,11,22,57,127,221,1
36
150 DATA 192,32,160,208,28,252,
222,138
151 DATA 3,6,7,6,3,1,65,37
152 DATA 96,48,112,48,96,64,64,
210
153 DATA 19,11,11,41,41,165,165
,255
154 DATA 228,232,232,200,200,20
0,210,255
155 DATA 255,238,238,68,68,68,0
,0
156 DATA 0,0,68,68,68,238,238,2
55
160 FOR n=USR "a" TO USR "u"+7:
  READ x: POKE n,x: NEXT n
170 FLASH 0: OVER 0: INVERSE 0:
  BORDER 1: PAPER 1: INK 7: BRIGH
T 1: CLS
180 LET a$="1100010001101010101
01010100011001100101010001010101
0010001101010000000000000000011
10010010001000100010101000100011
00111010001000100010101110111"
190 LET p=1
200 FOR n=1 TO 9: FOR m=1 TO 16
210 IF a$(p)="1" THEN PRINT I
NK 6:AT n*2,(m-1)*2:"FG":AT n*2+
1,(m-1)*2:"HI"
215 IF INKEY$="s" THEN LET m=1
00: LET n=200: NEXT m: NEXT n: G
O TO 400

```

Infinite screens, and all different, should keep you glued to this *Boulder Dash* variation. The idea of the game is to collect five diamonds on each screen, whilst avoiding Egbert, the alien. Whenever you tunnel underneath a rock, the rock will fall, either stunning the alien for a bonus, or destroying a diamond (not advised!). Every five screens there will be no earth to tunnel through and on later screens the alien will speed up.

When typing the listing, you'll come across underlined characters. These are in place of the user-defined graphics which are normally scattered through the listing and should be typed in GRAPHICS mode (the G cursor). So if you see the character F, enter graphics mode (CAPS SHIFT + '9'), then hit the character and then return to normal mode (CAPS SHIFT + '9' again). You may also come across a set of characters such as sh8. These refer to Spectrum symbol graphics (above the keys 1 to 8). Again in graphics mode, hold down the SHIFT key whilst pressing the number (in this case '8'). When the program has been run, a listing will show the graphics, shown in the diagram below.



```

220 LET p=p+1
230 NEXT m: NEXT n
240 FOR n=0 TO 15
245 IF INKEY$="s" THEN GO TO 4
00

```

# SPECTRUM 16/48K

```
250 PRINT AT 0.n: INK 5:" BC":A
T 1.n: INK 4:" DE"
260 PRINT AT 11.30-n: INK 7:"NO"
"
270 PRINT AT 20.n: INK 3:" JK":
AT 21.n:" LM"
280 BEEP .01.n: NEXT n
290 LET a$="
```

Press S to play.

A.Clapham presents

-----ROCKFALL-----

Guide Ernie the mining robot to collect five diamonds. Avoid

Eggbert NO and do not run o  
ut of air. KEYS Q=

UP A=DOWN M=RIGHT N=LEFT

```
"
295 LET a$=a$+"5 points for eac
h jewel + bonus for screen and k
illing Eggbert. CAN YOU BEAT
THE HIGHSORE OF "+h$+" ???
"
```

```
300 DIM k(4): LET c=1: LET d=1:
RESTORE 370: FOR n=1 TO 4: READ
k(n): NEXT n: RESTORE 360: READ
q.z: IF INKEY$="s" THEN GO TO
400
```

```
310 LET x=0: FOR n=1 TO LEN a$-
32: LET x=x+1: IF x=8 THEN LET
x=1
```

```
320 IF INKEY$="s" THEN LET n=1
000: GO TO 340
```

```
330 PRINT AT 10.0: INK x:a$(n T
O n+31): BEEP .01.q: LET z=z-1:
IF z=0 THEN READ q.z: IF z=99 T
HEN RESTORE 360: READ q.z
```

```
335 BEEP .01.k(c): LET d=d+1: I
F d=3 THEN LET d=1: LET c=c+1:
```

```
IF c=5 THEN LET c=1
```

```
340 NEXT n: IF n<1000 THEN GO
TO 310
```

```
350 GO TO 400
```

```
360 DATA -59.16.4.3,-59.1.4.4.7
.3,-59.1.7.4.11.4.9.2.7.2.11.8.1
6.3,-59.1.16.2.14.2.11.4.9.2.7.2
.11.4.9.2.7.2.4.4.4.8
```

```
365 DATA 4.4.99.99
```

```
370 DATA -8,-1.0.-1
```

```
390 DATA 99.99
```

```
400 LET sc=1
```

```
410 LET s=0
```

```
420 LET l=3
```

```
430 INK 7: PAPER 1: BRIGHT 1: C
LS
```

```
440 PRINT AT 10.10:"SCREEN ":sc
```

```
441 IF sc=1 THEN PRINT AT 12.0
```

```
: PAPER 7: INK 0:" THE EA
```

```
SY SCREEN "
```

```
442 IF sc=5 THEN PRINT AT 12.0
```

```
: PAPER 7: INK 0:" Mind the V
```

```
enus Fly Traps "
```

```
443 IF sc=11 THEN PRINT AT 12.
```

```
0: INK 0: PAPER 7:"Watch out.you
```

```
go half the speed!"
```

```
444 IF sc=16 THEN PRINT AT 12.
```

```
0: PAPER 7: INK 0:" Mind th
```

```
e robot traps "
```

```
445 IF sc=18 THEN PRINT AT 12.
```

```
0: PAPER 7: INK 0:"You must have
```

```
skill to succeed!"
```

```
450 BEEP .2.4: BEEP .2.0: BEEP
```

```
.2.4: BEEP .4.7
```

```
460 RANDOMIZE sc
```

```
470 PAUSE 150: CLS : IF sc/5<>I
```

```
NT (sc/5) THEN LET ws="AA"+FN s
```

```
$(2)+"AAAAAAA"+FN s$(1)+"AAAAAA
```

```
"-FN s$(4)+"AAAA"-FN s$(6)+"AAAA
```



# ROCK FALL

BY ANDREW CLAPHAM



```

": FOR n=0 TO 21: PRINT AT n,0:
INK 2: PAPER 1:w$: NEXT n
480 IF sc>1 THEN DIM w$(32): F
RINT AT 10,0:w$:AT 11,0:w$
485 FOR m=NOT (sc>4) TO 2: FOR
n=1 TO 12+sc*m
490 LET x=INT (RND*10)*2: LET y
=INT (RND*16)*2: IF x=10 THEN G
O TO 490
495 IF m=0 THEN PRINT AT x,y:
INK 4: BRIGHT 0:"EQ":AT x+1,y:"R
S"
500 IF m=2 THEN PRINT AT x,y:
INK 6:"FG":AT x+1,y:"HI"
520 IF m=1 THEN PRINT AT x,y:
PAPER 1:" ":AT x+1,y:" "
530 NEXT n: NEXT m
540 FOR n=1 TO 5
550 LET x=INT (RND*10)*2: LET y
=INT (RND*16)*2
560 IF x=10 THEN GO TO 550
565 IF ATTR (x,y)=71 THEN GO T
O 550
570 PRINT AT x,y: INK 7: PAPER
0: BRIGHT 1:"JK":AT x+1,y:"LM":
NEXT n
580 LET h=10: LET p=0: LET oh=h
: LET op=p: LET mh=10: LET mp=30
: LET omh=mh: LET omp=mp
590 LET md=1
700 LET g=0: LET air=200-(sc*25
): IF air<60 THEN LET air=60
710 GO SUB 1000
720 GO SUB 2000
730 DIM x(4): DIM y(4): LET x(1
)=-2: LET y(2)=2: LET x(3)=2: LE

```

```

T y(4)=-2
740 IF sc<11 OR sc>14 THEN GO
TO 770
750 GO SUB 3000: GO SUB 2000: G
O SUB 3000: GO SUB 2000: GO SUB
4000: GO SUB 1005
760 GO TO 750
770 IF sc<16 OR sc>25 THEN GO
TO 800
780 FOR u=1 TO 4: GO SUB 2000:
GO SUB 3000: GO SUB 4000: GO SUB
1005: NEXT u
790 GO SUB 9000: GO TO 780
800 GO SUB 2000: GO SUB 3000: G
O SUB 4000: GO SUB 1005
810 GO TO 800
1000 PRINT 0:AT 1,0:"SCR ":s:TA
B 10:"LVS ":l:TAB 16:"SCRN ":sc:
TAB 24:"AIR ":air:AT 0,0:"
HIGHSCORE ":hs
1001 RETURN
1005 PRINT 0:AT 1,4:s:AT 1,28:a
ir:" "
1010 LET air=air-1: IF air<15 TH
EN BEEP .1,30
1020 IF air<=0 THEN FOR z=1 TO
15: BEEP .1,30: NEXT z: GO TO 60
00
1030 RETURN
2000 PRINT AT oh,op:" ":AT oh+1
.op:" ":AT h,p: INK 5:"BC":AT h
+1,p: INK 4:"DE":AT omh,omp:" "
:AT mh,mp: INK 7:"NO"
2010 LET oh=h: LET op=p: LET omh
=mh: LET omp=mp
2020 RETURN

```

# SPECTRUM 16/48K

```

3000 LET nmd=md+1: IF nmd=5 THEN
  LET nmd=1
3010 LET nmh=nh+y(nmd): LET nmp=
mp+x(nmd)
3020 IF nmh<0 OR nmh>20 OR nmp<0
  OR nmp>30 THEN GO TO 3500
3030 IF SCREEN$ (nmh,nmp)<>" " T
HEN GO TO 3500
3040 LET md=nmd: LET mp=nmp: LET
mh=nmh: GO TO 3600
3500 LET nmh=nh+y(md): LET nmp=m
p+x(md)
3510 IF nmh<0 OR nmh>20 OR nmp<0
  OR nmp>30 THEN GO TO 3550
3520 IF SCREEN$ (nmh,nmp)<>" " T
HEN GO TO 3550
3530 LET mh=nmh: LET mp=nmp: GO
TO 3600
3550 LET md=md-1: IF md=0 THEN
LET md=4
3600 IF ABS (omh-oh)<4 AND ABS (
omp-op)<4 THEN GO SUB 8500: GO
TO 6000
3610 RETURN
4000 LET nh=h+((INKEY$="a")-(INKE
Y$="q"))*2: LET np=p+((INKEY$="
m")-(INKEY$="n"))*2
4010 IF nh<0 OR nh>21 OR np<0 OR
np>31 THEN LET nh=oh: LET np=o
p
4020 IF ATTR (nh,np)=78 THEN BE
EP .01,12: LET nh=oh: LET np=op
4030 LET h=nh: LET p=np
4040 IF ATTR (h,p)=71 THEN FOR
n=1 TO 5: BEEP .01,n*5: NEXT n:
LET g=g+1: LET s=s+5: IF g=5 THE
N GO TO 8000
4045 IF ATTR (h,p)=12 THEN FOR
n=1 TO 5: BEEP .05,10: BEEP .05,
0: NEXT n: GO TO 6000
4046 IF ATTR (h,p)=85 THEN FOR
n=20 TO 0 STEP -1: BEEP .001,n:
NEXT n: GO TO 6000
4050 IF oh<>0 AND oh<>20 THEN I
F ATTR (oh-1,op)=78 THEN GO SUB
4500
4100 RETURN
4500 PRINT AT oh-2,op:" ":AT oh
-1,op:" ": FOR n=h TO 18 STEP 2
4510 PRINT AT n,op:" ":AT n+1,o
p:" ":AT n+2,op: INK 6:"FG":AT
n+3,op: INK 6:"HI"
4520 BEEP .01,(20-n)*2
4530 IF n+4=omh AND op=omp THEN
  FOR q=69 TO 0 STEP -5: BEEP .00
5,q: NEXT q: FOR q=0 TO 69: BEEP

```

```

.005,q: NEXT q: PRINT AT omh,om
p:" ": LET s=s+100: LET mh=0: L
ET omh=0: LET mp=14: LET omp=14
4540 IF n<18 THEN IF SCREEN$ (
n+4,op)<>" " THEN GO TO 4600
4550 PRINT AT oh,op:" ":AT oh+1
,op:" ":AT h,p: INK 5:"BC":AT h
+1,p: INK 4:"DE"
4560 NEXT n: LET n=18
4600 PRINT AT n+2,op: INK 6:"FG"
:AT n+3,op: INK 6:"HI"
4610 IF n+2=oh THEN LET oh=h
4620 RETURN
6000 PRINT AT oh,op:" ":AT oh+1
,op:" ":AT h,p: INK 5:"sh2sh7":
AT h+1,p:"sh2sh7": FOR n=69 TO 0
STEP -2: BEEP .01,n: NEXT n
6010 LET l=1-1: IF l=0 THEN GO
TO 7000
6020 GO TO 430
7000 BEEP .5,0: BEEP .5,0: BEEP
.25,0: BEEP .5,0: BEEP .5,2: BEE
P .25,1: BEEP .5,1: BEEP .25,0:
BEEP .5,0: BEEP .25,-1: BEEP 1,0
7010 LET a$=" " G A M E O
V E R "
7020 FOR n=1 TO LEN a$: PRINT AT
10,n-1:a$(n): BEEP .005,60: BEE
P .005,10: NEXT n
7030 BEEP .5,-48
7035 IF s>VAL h$ THEN LET h$=ST
R$ s
7040 PAUSE 200: GO TO 170
8000 BEEP .2,12: BEEP .2,7: BEEP
.2,9: BEEP .4,11: BEEP .1,9: BE
EP .3,11: BEEP .4,12
8010 FOR n=air TO 0 STEP -3: LET
s=s+3: BEEP .003,n/4: PRINT )0:
AT 1,4:s:AT 1,28:n:" ": NEXT n:
LET sc=sc+1
8020 GO TO 430
8500 LET dy=omh*8-h*8: LET dx=om
p*8-p*8: FOR y=175-h*8 TO 175-h*
8-8 STEP -2: PLOT p*8+8,y: DRAW
dx,-dy: BEEP .01,RND*50: NEXT y:
RETURN
9000 LET x=INT (RND*16)*2: LET y
=INT (RND*11)*2
9010 IF ATTR (y,x)=71 OR (y=mh A
ND x=mp) OR (y=h AND x=p) THEN
RETURN
9020 PRINT AT y,x: INK 5: PAPER
2:"TT":AT y+1,x:"UU"
9030 RETURN
9999 DEF FN s$(s)=("AA") AND sc<=
s)+(" " AND sc>s)

```

# THE GREAT C&VG COVER-UP!



Why not brighten up those boring old blank tape labels by using *C&VG's* Book of Games cassette inlays? Just use a sharp pair of scissors to cut out the labels, paste them onto the blank cassette inlay and you'll have your very own custom-made *C&VG* games cassette. Neat eh? If you don't want to spoil your Book of Games, why not get together with a friend and swap them — or else buy another copy of *C&VG*? We have to keep paying protection money to the Bugs somehow!